Summer 1980 Vol. 2 No.1 $2.50 U.S.A.
A McGRAW-HILL Publication

GUIDE TO PERSONAL COMPUTING

onComputing™

14151

Word Processors
Choosing the Right One For You

Jerry Pournelle: Living With Computers

Reviews: Atari 800 APF's Imagination Machine

Nestar's Multiuser Computer Network

"The Source": Big Computer Power in Your Home

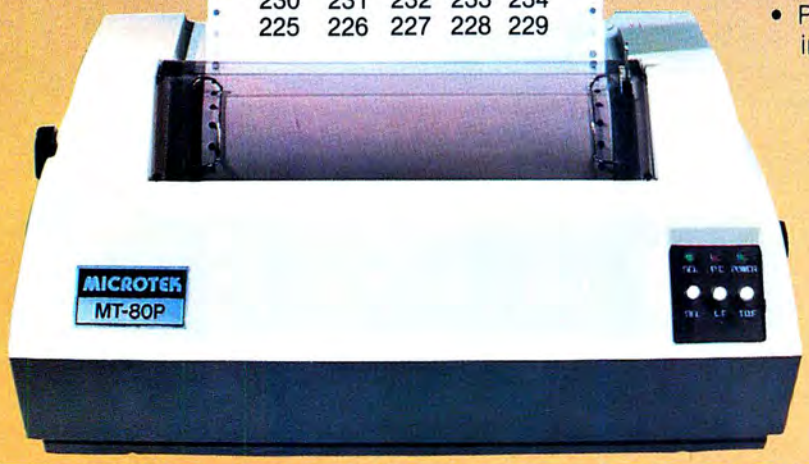
Summer 1980 onComputing Volume 2 Number 1

MICROTEK

SO RELIABLE WE GIVE YOU

365 DAYS WARRANTY

360 361 362 363 364
355 356 357 358 359
350 351 352 353 354
345 346 347 348 349
340 341 342 343 344
335 336 337 338 339
330 331 332 333 334
325 326 327 328 329
320 321 322 323 324
315 316 317 318 319
310 311 312 313 314
305 306 307 308 309
300 301 302 303 304
295 296 297 298 299
290 291 292 293 294
285 286 287 288 289
280 281 282 283 284
275 276 277 278 279
270 271 272 273 274
265 266 267 268 269
260 261 262 263 264
255 256 257 258 259
250 251 252 253 254
245 246 247 248 249
240 241 242 243 244
235 236 237 238 239
230 231 232 233 234
225 226 227 228 229



TOTAL PEACE OF MIND.

We are business people, just like you. We wouldn't stick our neck out unless we were certain we could back our claims and promises to the last letter. We wouldn't offer a legally binding long-term warranty on our Microtek MT-80, unless we were positive of its solid reliability.

We are sure that our versatile, alphanumeric line printer will provide you consistent, dependable performance. This is why we give you, not the usual 90-days in fine print, but an incredible 365-days warranty. We stand behind every product we make.

LOADED WITH INNOVATIONS

- 40, 80 or 120 columns (software selectable)
- Non-thermal paper, pin feed
- 125 CPS, 70 lines per minute
- 9 × 7 dot matrix
- Vertical format unit
- 96-character ASCII (upper and lower case)
- Adjustable forms width to 9½"
- Parallel and serial (RS-232C) interfaces available

OUR UNIT PRICE
$795 Parallel
$895 Serial (RS-232C)

RELIABILITY

For more information contact:
MICROTEK, Inc.
9514 Chesapeake Dr.
San Diego, CA 92123
Tel. (714) 278-0633
TWX 910-335-1269

Circle 210 on inquiry card.

Solve your personal energy crisis. Let VisiCalc™ Software do the work.

With a calculator, pencil and paper you can spend hours planning, projecting, writing, estimating, calculating, revising, erasing and recalculating as you work toward a decision.

Or with the Personal Software™ VisiCalc program and your Apple* II you can explore many more options with a fraction of the time and effort you've spent before.

VisiCalc is a new breed of problem-solving software. Unlike prepackaged software that forces you into a computerized straight jacket, VisiCalc adapts itself to any numerical problem you have. You enter numbers, alphabetic titles and formulas on your keyboard. VisiCalc organizes and displays this information on the screen. You don't have to spend your time programming.

Your energy is better spent using the results than getting them.

Say you're a business manager and want to project your annual sales. Using the calculator, pencil and paper method, you'd lay out 12 months across a sheet and fill in lines and columns of figures on products, outlets, salespeople, etc. You'd calculate by hand the subtotals and summary figures. Then you'd start revising, erasing and recalculating. With VisiCalc, you simply fill in the same figures on an electronic "sheet of paper" and let the computer do the work.

Once your first projection is complete, you're ready to use VisiCalc's unique, powerful recalculation feature. It lets you ask "What if?," examining new options and planning for contingencies. "What if" sales drop 20 percent in March? Just type in the sales figure. VisiCalc instantly updates all other figures affected by March sales.

Circle 10 on inquiry card.

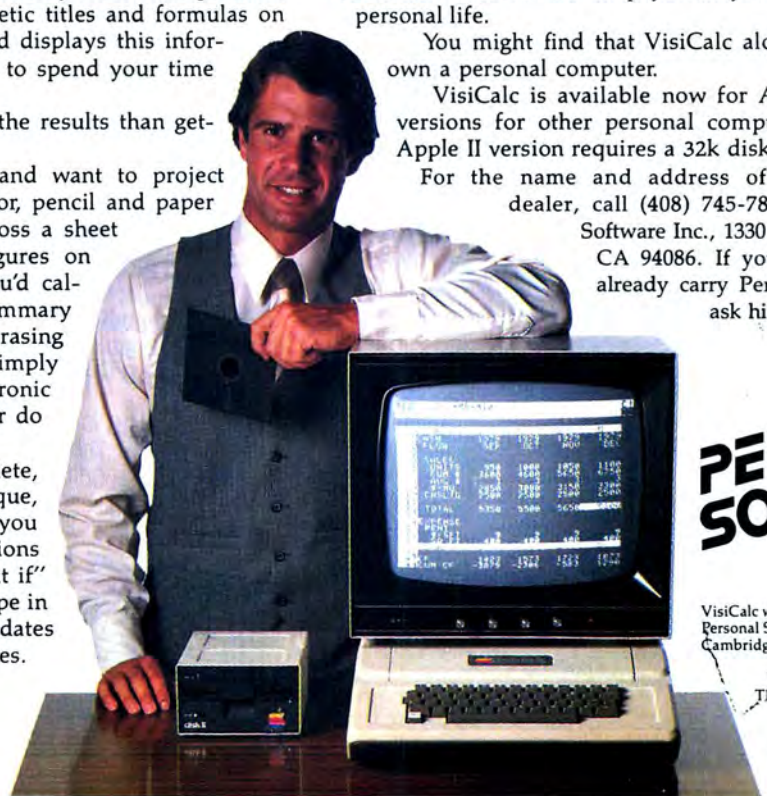
Or say you're an engineer working on a design problem and are wondering "What if that oscillation were damped by another 10 percent?" Or you're working on your family's expenses and wonder "What will happen to our entertainment budget if the heating bill goes up 15 percent this winter?" VisiCalc responds instantly to show you all the consequences of any change.

Once you see VisiCalc in action, you'll think of many more uses for its power. Ask your dealer for a demonstration and discover how VisiCalc can help you in your professional work and personal life.

You might find that VisiCalc alone is reason enough to own a personal computer.

VisiCalc is available now for Apple II computers with versions for other personal computers coming soon. The Apple II version requires a 32k disk system.

For the name and address of your nearest VisiCalc dealer, call (408) 745-7841 or write to Personal Software Inc., 1330 Bordeaux Dr., Sunnyvale, CA 94086. If you favorite dealer doesn't already carry Personal Software products, ask him to give us a call.



PERSONAL SOFTWARE

VisiCalc was developed exclusively for Personal Software by Software Arts, Inc., Cambridge, Mass.

TM—VisiCalc is a trademark of Personal Software, Inc.

*Apple is a registered trademark of Apple Computer, Inc.

TBS-80 DATA PROCESSING SYSTEMS. ONE STEP BEYOND.

If you thought the TRS-80™ microcomputer was just a toy, think again. These **TBS-80** software systems will turn that computer into a **powerful data processor.**

INFORMATION SYSTEM by Dale Kubler is simply the best in-memory, data base manager on the market. It allows you to create files with up to ten fields per record, up to 40 characters per field and 200 characters total per record. Data from the keyboard is entered directly onto a screen display of one entire file. Once entered, you can sort or search your entire data base by any category and have the information desired displayed on the screen. **INFORMATION SYSTEM** provides a thorough editing mode allowing changes by line without rewriting an entire file. This program allows you to program your own printouts to almost any form you desire for line or serial printers. Screen prints from anywhere in the program are also available **INFORMATION SYSTEM** creates either disk or cassette files depending upon the version you use. From mail lists to recipes, this program is the ideal small system information manager. The price for this program, 32K up disk is $34.50. For systems 16K up tape it's $24.50.

DATA MANAGER by Dale Kubler starts out where INFORMATION SYSTEM leaves off. Requiring 32K and one disk, it accepts up to ten user-defined fields with up to forty characters per field and 255 characters per record. As with all TBS software, data entry and editing is professional and simple to use. What makes this program stand apart from "in-mem" data managers is that it uses up to four disks on line as memory, or as much as 320K of memory storage. Because disk sorts take more time than in-mem sorts, **DATA MANAGER** enables the user to create and maintain up to 5 "key" sort files for quick access of data. A utility program is provided to calculate the number of records possible since the amount of records you can maintain is dependent on a number of variables. This program also supports the upper/lower case modification, and printouts can be programmed to almost any format and sent to line or serial printer. *Background* printing is provided enabling the computer to search and print at the same time. If you already have INFORMATION SYSTEM, **DATA MANAGER** will accept those files. A necessity for organized people, this program sells for $49.50.

BUSINESS MAIL SYSTEM by Dale Kubler is designed for large-scale business users. Requiring 32K, two disks and printer, this program will store up to 150,000 names in a single file spread out over multiple disks. Each data disk holds 500 names. After data entry, BMS automatically sorts the data by zip code and alphabetical order within the zip code. The program tells you when and which data disk to insert, expanding your files automatically until you've reached 300 disks. Data is input directly onto formatted screen display with the option to use Company Name/Attention instead of Last Name/First Name. Three numeric and one alpha code fields are provided to help you use the search and printout mode. **BUSINESS MAIL SYSTEM** allows you to program the number and spacing of your labels. With more features than can be described here, this high-powered program sells for $125.00.

TEXT MERGE is the program that puts it all together. If you have the ELECTRIC PENCIL from Michael Shrayer, 32K and one disk drive, then this program is a must. It will merge your data base from any of the above programs with an Electric Pencil file. For example, when you write a letter that is going to several hundred people, you can "code" it by entering a field name from the above programs in place of the actual information. Then, when **TEXT MERGE** is run, it will print out your Pencil file and substitute the "code" with the actual data. In other words, you can print out 1,000 personalized letters without stopping the computer. This program will also enable you to selectively search out only the records from your data base that you wish to use. Also included is the ability to set left, right, top and bottom margins, set page numbers anywhere on the page, and print out right justified if you so choose. **TEXT MERGE** will turn your computer into a powerful data processor and it sells on disk for $49.50.

TBS has other incredible software for Tandy's microcomputer. Intent on making it a powerful tool, we have **large scale business accounting systems, general accounting systems, system utilities** and the **Library 100.** We have the only **DISK HEAD CLEANER** (for APPLE too!) and **GRAN MASTER DISKETTES,** the best on the market.

TBS is **YOUR COMPANY,** and we build systems, not just software. The above products are available now, nationwide. Visit your local Computer Dealer or Associate Radio Shack Store and demand the best, demand TBS. For more information, contact us through the numbers below.

™TRS-80 is a trademark of the Tandy Corporation.

TBS™

THE BOTTOM SHELF, INC.

(404)939-6031 • P.O. Box 49014 • Atlanta, GA 30359

Circle 20 on inquiry card.

Illustration by Sam Petrucci of Gunn Associates, Boston MA

Vol. 2 No. 1
Summer 1980

onComputing

onComputing is published four times a year by onComputing Inc, 70 Main St, Peterborough NH 03458, a wholly-owned subsidiary of McGraw-Hill, Inc. Address all mail except subscriptions to above address: phone (603) 924-9281. Address all subscriptions, change of address, USPS Form 3579, and fulfillment questions to onComputing, POB 307, Martinsville NJ 08836. Controlled circulation pending, Waseca MN 56093 and Peterborough, NH 03458. (ISSN 0194-8075) Subscriptions are $8.50 for one year in USA and its possessions. In Canada and Mexico $10 for one year. Other foreign countries $12 for one year surface delivery. Single copy price is $2.50 in the USA and its possessions, $2.90 in Canada and Mexico, $4 in Europe, and $4.50 elsewhere. Foreign subscriptions and sales should be remitted in United States funds drawn on a US bank. Printed in the United States of America. **Subscription WATS Line (800) 258-5485**
Address all editorial correspondence to the editor at 70 Main St, Peterborough NH 03458. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos.

PUBLISHERS:
Gordon R Williamson
Virginia Londoner

ASSOCIATE PUBLISHER:
John E Hayes

EDITORIAL DIRECTOR:
Carl T Helmers, Jr

EDITOR-in-CHIEF:
CHRISTOPHER P MORGAN

PRODUCTION and DESIGN:
Nancy Estle
Ellen Klempner

CIRCULATION MANAGER:
Gregory Spitzfaden

ADVERTISING DIRECTOR:
Tom Harvey

NATIONAL ADVERTISING SALES REPRESENTATIVES:
HAJAR ASSOCIATES INC

East:
280 Hillside Av
Needham Heights MA 02194
(617) 444-3946

521 Fifth Av
New York NY 10017
(212) 682-5844

West, Southwest:
1000 Elwell Ct
Suite 227
Palo Alto CA 94303
(415) 964-0706
or (714) 540-3554

Midwest:
2405 Lawndale
Evanstown IL 60201
(312) 864-3467

Officers of McGraw-Hill Publications Company:
Paul F. McPherson, President; Executive Vice Presidents: James E. Boddorf, Gene W. Simpson; Group Vice President: Daniel A. McMillan; Senior Vice President-Editorial: Ralph R. Schulz; Vice Presidents: Kemp Anderson, Business Systems Development; Stephen C. Croft, Manufacturing; Robert B. Doll, Circulation; James E. Hackett, Controller; William H. Hammond, Communications; Eric B. Herr, Planning and Development; John W. Patten, Sales; Edward E. Schirmer, International.
Officers of the Corporation:
Harold W. McGraw Jr., President, Chief Executive Officer and Chairman of the Board; Robert F. Landes, Senior Vice President and Secretary; Ralph J. Webb, Treasurer.

Editor's Message

Taming the Wild Syntax

What's all the fuss about word processors? And, by the way, what *is* a word processor? We'll try to answer both of these questions in this issue of onComputing.

During the past year or so, advertisements have flooded the personal computer media for word processors of every type and description. Put simply, a word processor is a computer program that lets you go from an idea to the printed page quickly and efficiently — whether you're writing a letter, a novel, a report, or whatever.

For our part, we hope to sort out some of the confusion about word processors, text editors, formatters, and the rest of the word-processing jargon in articles by Larry Press and Jerry Pournelle.

Larry has been working on this project for nearly a year, in conjunction with his company, the Small Systems Group of Santa Monica, California. Out of this research has come a detailed evaluation of four word processors: Auto Scribe, Electric Pencil, Magic Wand, and WordStar.

Jerry Pournelle's name will be familiar both to onComputing readers (he wrote an article about writing with a computer in our Summer 1979 issue) and to science fiction readers. Jerry's books have been on the best-seller lists many times. After writing two million words of fiction and nonfiction with various word processors during the past few years, he's become an expert by osmosis.

For do-it-yourselfers, we've included a simple (albeit *slow*) line editor for the Radio Shack TRS-80 personal computer written in BASIC. The program is obviously no match for the sophisticated editing programs on the market, but it *is* a good way to learn how some of them work.

* * *

Also in This Issue

The Source has received much publicity lately in the *Wall Street Journal* and other media. For a small hourly fee, subscribers can link their personal computers or terminals to a central, time-sharing computer over telephone lines and receive the UPI wire service and the New York Times data base, as well as hundreds of programs. Is it worth it? Read Ken Skier's incisive review and find out.

The Nestar System previewed in this issue (see "Nestar Lets Personal Computers Talk to Each Other") lets up to sixty-four computers (Apples, in this case) talk to one another and share programs and data on the same bus. (A *bus* is a set of wires that transmit data and addresses within a computer, or, in this case, between computers.)

Elsewhere in this issue, we cover floppy-disk drives for the beginner, computer animation as a tool for teaching children along with a review of APF's new personal computer, and much more.

You might have noticed that we've revamped the look of the magazine somewhat. Both the cover and the interior layout have been modernized by Nancy Estle and Ellen Klempner. We hope you like the changes, and we look forward to your comments.

Chris Morgan
Editor-in-Chief

"INTRODUCING THE HOTTEST THING OFF THE DRAWING BOARDS: A COMPUTER TRIO THAT TAKES YOU FROM SMALL TO BIG."

Geof Karlin
Director of Systems Development

If you're considering your first computing system, you need to know more than what it will do for you today. You need to know what it will do tomorrow.

Many businesses outgrow their first computer within a year or two. And when they do, they find their system is difficult if not impossible to add onto–because adding on requires a different manufacturer's equipment, different operating system, different programming languages.

At ADDS, we've just solved this problem with Multivision, a compact trio of stacking computers with ADDS' CP/M®–compatible, multi-user operating system. You can begin with one and expand as needed.

MULTIVISION 1 (top module) is a get-started computer with 5 MHz processor, 64K bytes of RAM (Random Access Memory) and floppy-disk storage capacity of 700K bytes. It's available with a wide range of business application software. We even offer our own word processing package. A fully loaded Multivision 1 is list-priced at $3,785 without terminal.

MULTIVISION 2 (top and bottom modules) uses the new Winchester technology to provide 5M or 10M additional bytes of hard-disk storage. List-priced at $7,995 for 5M bytes of disk, it is thousands less than other hard-disk systems.

MULTIVISION 3 (entire stack) turns your computer into a multi-user system with up to 256K bytes of RAM that supports up to four display terminals. It allows four different parts of your business–i.e., accounting, marketing, purchasing, and shipping–to share data and simultaneously use the system. No other microsystem offers so much for so little.

Before you decide upon any small computer, look into ADDS Multivision. For years we've been the largest supplier of display terminals to computer giants.

Now we're making a system for *you*.

For information, write: Systems Division, Applied Digital Data Systems Inc., 100 Marcus Boulevard, Hauppauge, N.Y. 11787. Dealer inquiries invited.

CP/M is a registered trademark of Digital Research, Inc.

ADDS MULTIVISION

ADDS
Applied Digital Data Systems Inc.

SOMETHING EXTRA IN EVERYTHING WE DO

Circle 30 on inquiry card.

Letters Letters

Left-Wing Computers?

Dear Editor:

Having recently read the Spring 1980 onComputing, I felt I had to write and say how much I disliked "An Apple in Hanoi," by Everett Hafner. The actions of the author bordered on treason, and you should be ashamed to have printed it. Is yours a left-wing magazine? If not, that's surely how it looks.

E E Carcen
Santa Barbara CA

Everett Hafner replies:

I very much appreciate the concern that readers like Mr Carcen have expressed about the activities that Western scientists engage in with scientists of Socialist block countries. My article was not meant to imply any political alliance between North Vietnamese scientists and myself. In fact, none of the people I met in Vietnam had any sympathy for the aggressive policies of their country.

I believe that peaceful scientific interaction is one important way to bring countries closer together and avoid aggression.

Voicing an Opinion

Dear Editor:

In the article "A Voice for Bill" by Mr Mark Dahmke in your Winter 1979 issue, there are a number of inaccurate and false representations concerning the Votrax Division of Federal Screw Works and its products. I would like to set the record straight for your readers.

Votrax does in fact manufacture a talking communication aid for nonspeaking people. It is called the Phonic Mirror HandiVoice. The two models were introduced in 1977 and are distributed and serviced throughout the United States and Canada by HC Electronics Inc (a subsidiary of American Hospital Supply Corporation). The instrument comes in two models, with 500 or 1000 prestored words, phrases, and phonemes. The microprocessor-based instrument is battery operated and completely portable. Each model currently retails for $2195.

Mr Dahmke refers to an "$11,000 communications machine," by which I assume he means the nonportable institutional model of HandiVoice which was introduced in 1976, not 1975, as Mr Dahmke erroneously states. The system included a 400-word stored vocabulary, cabinet, power supply, speaker, and two access devices (terminals). Mr Dahmke later states he purchased a $400 synthesizer "similar in performance," but neglects to say that this synthesizer was only a board, no power supply, no cabinet, no vocabulary, no amplifier, no speaker.

Mr Dahmke refers to "negotiations" between Dr Lois Schwab of the University of Nebraska and Votrax. Votrax has never negotiated with or stated an intention to negotiate with Dr Schwab, nor has she declared an intention to negotiate with Votrax for the development of a communications aid. We have discussed, and continue to discuss, a number of major development projects with Dr Schwab because of her great knowledge and concern for disabled people. One such development project resulted in the portable Phonic Mirror HandiVoice discussed above. Votrax relies heavily on input from authorities like Dr Schwab; it is only with the interest of experts in several fields from across the country that useful communication aids can be developed.

There are a number of false implications in Mr Dahmke's article, such as the suggestion that his rather modest student design project is comparable to a two-year development effort by Votrax resulting in a finished, readily available product supported in the field, but I will not belabor these. It is sufficient to

say that the bulk of Mr Dahmke's representations about the Votrax Division and its products are false by statement or by implication. This information was not supplied by Votrax, nor did he confirm this information with Votrax prior to publishing the article.

Votrax continues to be interested in developments of communication aids for the handicapped. In that vein, we commend Mark Dahmke for his contributions to Mr Bill Rush. However, his overt or implied criticisms of Votrax in no way promote his project nor lessen the fact that Votrax does manufacture portable talking communication aids readily available to nonspeaking individuals.

Leonard H Magnuson
Votrax Division
Troy MI

Mark Dahmke replies:

When I said that the $400 Computalker board was comparable in performance to the Votrax $11,000 communication machine, I was referring only to speech quality and clarity. My statements concerning the price of the Computalker board were not misleading, since it is clearly stated in the article that $3000 was allocated to build the Bionic Voice system for Bill Rush. When the computer, power supply, amplifier, speaker, external mass storage device, keyboard, and either single-line display or full video display are added to the $400 Computalker board, the current price for my system runs from $4000 to $5600 with speech quality comparable to the $11,000 Votrax model that was available in 1976.

Mr Magnuson apparently underestimates my involvement in the nonvocal, nonverbal communications field. The project described in my article was not a mere student design project, nor was it a class project. I operate a computer consulting business and am involved in a wide range of ongoing consulting activities. I am currently developing a fully portable battery operated version of the Bionic Voice which will have a 40-character alphanumeric display (the HandiVoice has only a 3-digit numeric display), a full keyboard (or any variation desired by the user), plug-in memory modules with up to 1000 prestored words and phrases each, and room for 800 thru 1000 user-defined words and phrases that can be stored in internal memory that retains data when the unit is turned off. Following the original design of the Bionic Voice, the portable unit will operate in the scan mode or in full English. Unlike the HandiVoice, the Bionic Voice will also operate as a portable computer terminal and as a Teletype for the deaf. By interchanging plug-in modules, it may also function as an educational aid, with practice and drill programs to reinforce classroom education. This version will be comparable in price to the HandiVoice.

My philosophy has been and will continue to be that I market only equipment that has been customized to meet the needs of the user. It is also important that each unit be upward compatible with future needs of the user with a minimum of modifications and expense.

Mr Magnuson also states that no negotiations ever took place between Votrax and Dr Lois Schwab. Although there may have been a misunderstanding of the term "negotiations," he states that discussions have gone on and that development projects have resulted from these "discussions."

Mr Magnuson claims that I did not confirm the information about Votrax and its products. On the contrary, all of my information came from their sales literature and from demonstrations of the Votrax voice synthesizer product line.

onComputing: Pro

Dear Editor:

I have had previous experience

Letters continued on page 94

A Beginner's Guide

Understanding Floppy Disks

by Pamela Valentine

If there's one obstacle to faster acceptance of the personal computer as a valuable tool, it's the aura of mystery about its components, programming languages, and accessories. Understanding these pieces of the personal-computer puzzle is made even more difficult because technical jargon, as unique as any foreign language, surrounds them. (The first time I heard the expression "nested loop," I thought that it was meant to describe a bird showing off on its return home.)

And so it is with disk drives. The very name conjures up thoughts of a company's large data-processing operation, complete with raised floors, controlled temperatures, and stacks of printouts. But do disk drives belong only in the computer room? Not any more, as increasing numbers of personal computer users are switching from tape cassette units to disk drives, which provide substantially higher performance at relatively low cost.

The Data Phonograph

Understanding the principle

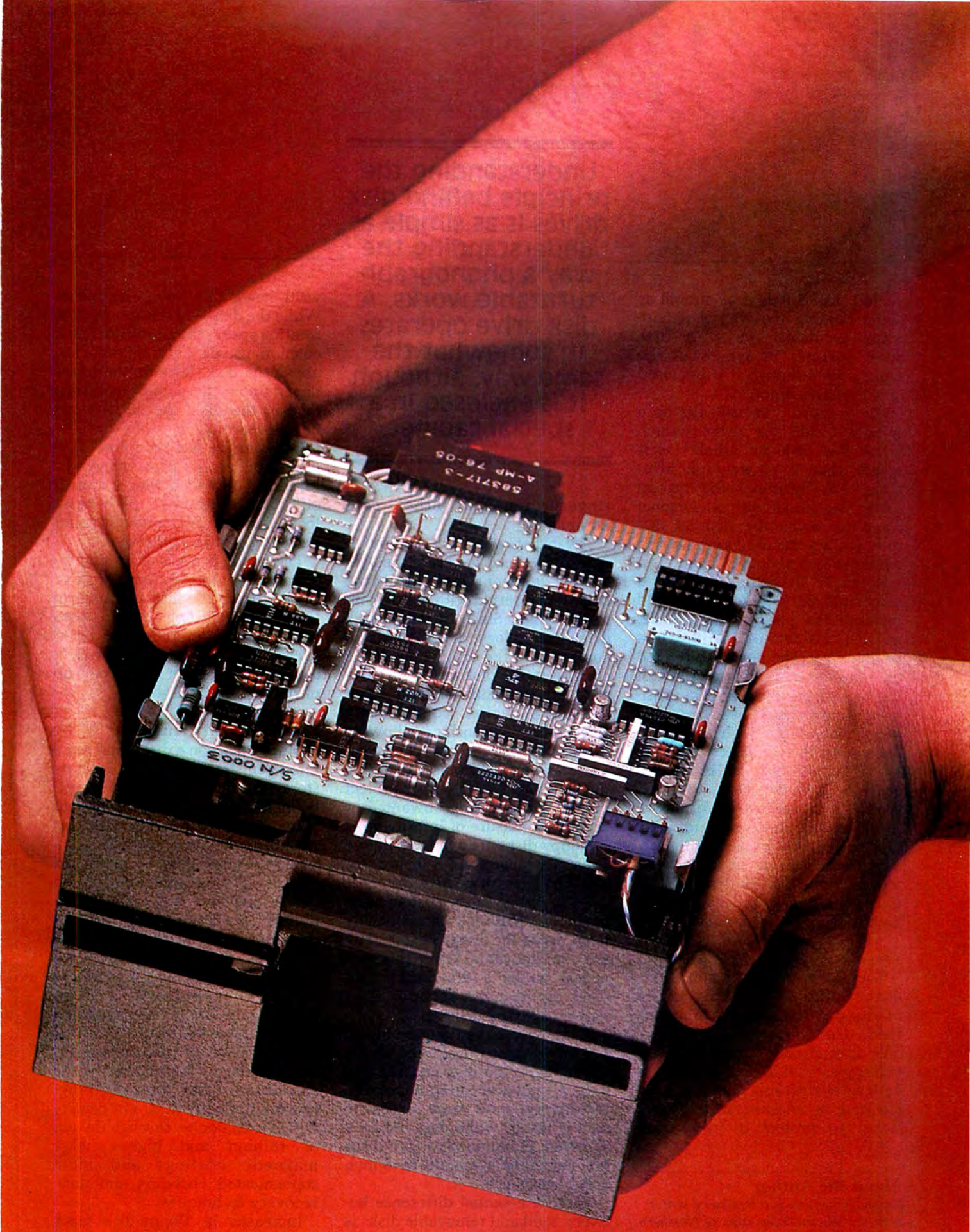


Photo 1: *The 5-inch floppy-disk drive with cover removed. Floppy disks are placed in the slot at front.*

Understanding the principle behind disk drives is as simple as understanding the way a phonograph turntable works. A disk drive operates in somewhat the same way, although it is enclosed in a special cabinet.

behind disk drives is as simple as understanding the way a phonograph turntable works. A disk drive operates in somewhat the same way, although it is enclosed in a special cabinet.

The "record," or *disk*, fits on a spindle inside the drive, which turns the disk at much faster speeds than a phonograph turntable. For example, the typical drive associated with a personal computer spins the disk at 300 rpm, while a standard long-playing album is turned at 33.3 rpm.

Within a disk-drive system, a record/playback (read/write) head replaces the phonograph needle, and it reads data on the disk in a similar way. The head is mounted on a *positioner*, the counterpart of the record player arm. The positioner moves the head laterally across the disk to any area requested by the computer program.

This capability is called *random access* and allows users to store, access, and retrieve information from the disk much more quickly than is possible by sequential access, which must be used with tape drives. The disk-drive system is powered by its own motor housed in the same cabinet.

The Data Record

There are two types of "records," or disks (the "media"), used in disk-drive systems: flexible and rigid. The flexible, or "floppy," disk is a circular piece of pliant mylar plastic. It is coated on one or both sides with a magnetic iron oxide and is encased in a plastic jacket to protect it from dust, fingerprints, and other contaminants.

The jacket, which has a cutout to permit the head to access the disk surface, is never removed from the disk. It is inserted with the disk into the drive. Since in floppy-disk drive technology, the head is in contact with the disk, some of the coating on the disk wears off. The soft, low-friction material that lines the jacket continuously cleans the disk surface, allowing the disk to rotate freely. The jacket also has an index hole for timing information and a write-protect notch for preventing unwanted or accidental erasure of information.

Floppy disks are available in two sizes, 5¼ inches and 8 inches, for use with their respective drives.

Rigid disks, also coated with magnetic material, are usually fixed inside the drive. That is, they are nonremovable. While this may appear to be a drawback, rigid disks are in fact a superior alternative to floppy disks.

A floppy disk can rotate only so fast before centrifugal force begins to misshape the disk and damage data. A rigid disk can spin far more rapidly and, as a result, access to data is faster, while data integrity is better maintained. Rigid disks are currently available in 8-inch and 14-inch sizes.

Another essential difference between rigid and removable disks is that, with the former, the read/write heads "fly" over the media. While the disk is at the appropriate rotating speed, the heads float on a cushion of air only microinches above the disk. Because the heads never touch the data-bearing surface of the disk, the surface can never wear out.

Both flexible and rigid disks contain concentric flat rings or "tracks," on which data is recorded. These tracks, physically located on the disk surface, cannot be seen by the unaided eye.

The actual process of writing information onto the floppy disk or reading from the disk is similar to that of recording music onto magnetic tape or playing it back, except that the read/write head in a disk drive can get to any portion of the disk in far less time.

Track and Bit Densities

More tracks on a disk—and more bits per track—provide greater storage capacity. More data capacity means less cost per bit stored. Since 1956, when IBM first announced rigid-disk storage technology, better disk surfaces and magnetic coatings, improved heads, and new data-coding techniques have resulted in significant increases to rigid-disk track and bit densities.

Because floppy-disk technology wasn't developed until 1971—again by IBM—increased track and bit densities for floppy disks have yet to be developed to their fullest capacities. It is estimated that before long, however, floppy-disk bit density will at least double through the use of thinner and higher-energy magnetic coatings and more sophisticated encoding and data-recovery techniques.

Increases in floppy-disk track density will be achieved through

About the Author

Pamela Valentine is a free-lance writer from the Silicon Valley area of Northern California.

improvements in media and more sophisticated head positioning/actuating systems.

Single versus Double

Single-sided floppy-disk technology, used by most personal computer manufacturers, was first introduced by IBM in 1973. Three years later, IBM announced the double-sided floppy-disk drive, which allows the user to read and update both sides of the diskette on-line. In addition, this technology is capable of reading and updating single-sided disks, so upward systems growth is possible. Most microcomputer manufacturers view the double-sided floppy disk as the next logical step in disk storage for their systems, although it is still not the dominant method.

Disk Drives for the Personal Computer

All disk drives operate in much the same manner, but there are various shapes, sizes, and capabilities available, depending upon the application.

Predominant today in the personal computer industry is the 5-inch floppy-disk drive. Introduced in 1976 by Shugart Associates, it evolved as the result of market requirements for a very compact recording medium for both word-processing systems and personal computers. Because it uses removable, flexible media, it is practical, versatile, and, most importantly, affordable.

This floppy-disk drive, similar to its larger predecessors in design, consists of read/write and control electronics, a drive mechanism, the read/write head, and the track-positioning mechanism. These components interpret and generate control signals, move the read/write head to the desired track on the floppy disk, and read or write data. (See photo 1.)

Interfaced to the microcomputer through a controller and formatter, the 5-inch floppy-disk drive is simple to install and easy to operate. The controller, however, is not usually a physical part of the drive itself, although some drive manufacturers have begun incorporating "controller chips" (integrated circuits) on the printed-circuit board that contains the drive's control electronics.

The controller performs functions such as interpreting and executing status signals from the drive and the microcomputer; positioning the read/write head to the track specified by the microcomputer; formatting the data into the particular format chosen; serializing and deserializing data to and from the drive; and performing a variety of tasks to monitor and determine command, data transfer, and status conditions (referred to as "handshake" routines).

Media for Personal-Computer Disk Drives

The floppy diskette for the 5-inch floppy-disk drive is a scaled-down version of the 8-inch floppy disk. While the 8-inch floppy disk generally has a total of seventy-seven tracks per side, the 5-inch usually has about thirty-five (although this can vary somewhat between various brands), designed to hold the equivalent of 25 to 35 typewritten pages of information.

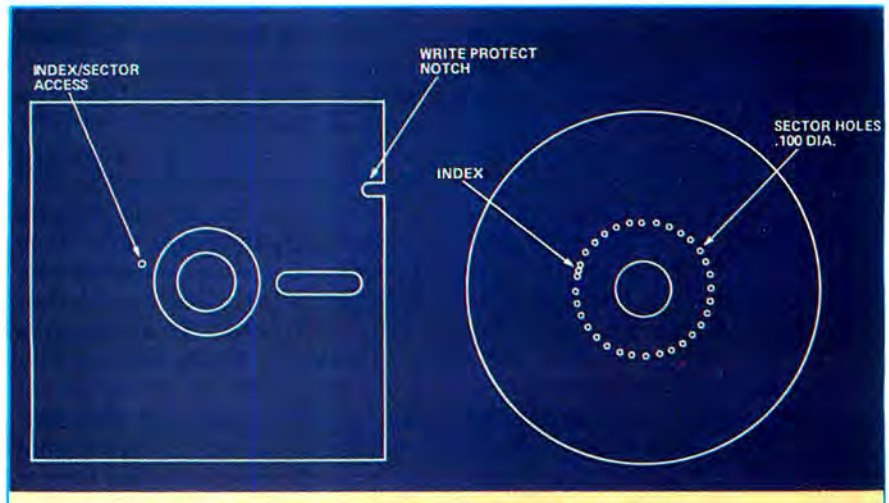


Figure 1: *The hard-sectored 5-inch floppy disk. The disk itself (at right) is normally kept inside its protective jacket (at left), even during use. The surface of the disk is divided into thirty-five concentric tracks, much like a phonograph record. In this hard-sectored disk, the tracks are divided into thirty-two sectors each. Sector positions are defined by a ring of concentric small holes shown here. The position of the holes can be read by the drive. The index hole gives the drive a reference position for determining where a particular track is. The write-protect notch on the jacket prevents any information on the disk from being erased when the notch is removed. The layout of the hard-sectored disk is in other respects similar to that of the soft-sectored disk shown in figure 2.*

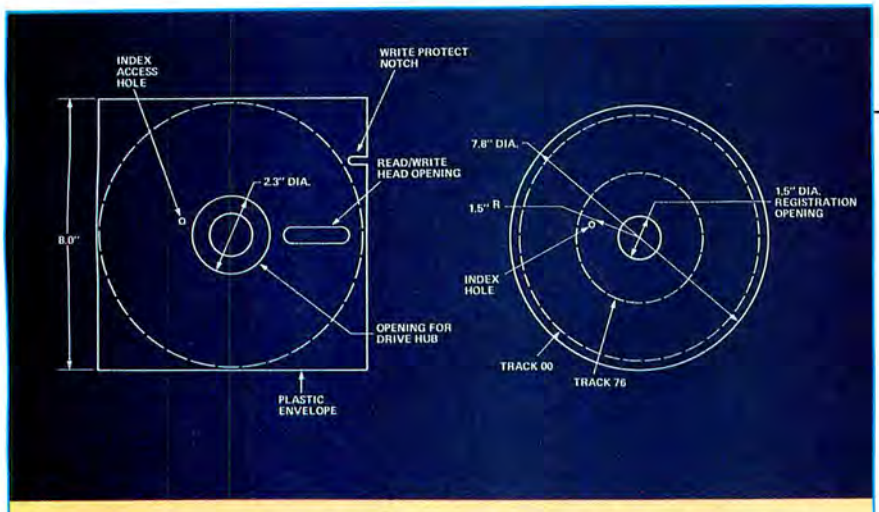


Figure 2: *The soft-sectored 5-inch floppy disk. Unlike the hard-sectored disk, the soft-sectored disk has no ring of many holes to determine sector position; it has only a single indexing hole. Sectors may be defined in any convenient way by the system, but a price is paid because information about the size of each sector must be stored in the sector itself, a process which robs a certain amount of storage space from the user. The advantage of soft-sectoring is flexibility; any number or length of sectors can be programmed up to the limit of the disk's capacity.*

The index hole in the diskette jacket allows light to pass to an optical detector in the drive. This causes an index-timing pulse to be generated on each revolution, signaling the beginning of a sector. With the standard 5-inch floppy diskette, rotating at 300 rpm, this pulse occurs every 200 microseconds (200 millionths of a second). The timing pulse is an important factor; if it is not accurate, data cannot be read by the drive.

Hard versus Soft Sectoring

The index pulse is also used to begin track formatting, which divides the tracks into fixed-length sectors or records. Sectoring is accomplished in either "hard" or "soft" sector recording format.

With hard-sectoring, the microcomputer may record up to thirty-two sectors per track. The diskette will contain an index hole, as discussed above, along with thirty-two equally spaced holes. Depending upon the application, the thirty-two sectors can be subdivided into sixteen or eight sectors in the sector separator/divide circuitry. With hard-sectoring, most of the work of sectoring is accomplished by the sector holes and sector divide electronics, so little software is required for record identification. (See figure 1.)

With soft-sectoring, the diskette contains only the index hole. Each record or soft sector must then be preceded by a unique recorded or written identifier. This earmarks the particular sector and its data field. Obviously, a premium of data space is required for soft-sectoring, which makes this format the least effective recording format relative to data capacity. However, flexibility is greater since any number or length of sectors can be programmed as long as the software and hardware are added for decoding. (See figure 2.)

The 8-Inch Floppy Disk

The smaller, 5-inch floppy-disk drives are more often found in personal-computer systems than the larger, greater-capacity, 8-inch units, but the 8-inch drives have always had a definite place in personal-computer systems. When MITS Inc launched the microcomputer industry with the Altair 8800 microcomputer kit, one of the first peripheral devices offered for use with the Altair was a dual-drive, 8-inch floppy-disk system manufactured by Pertec, the firm that later bought MITS. The popular CP/M operating system was developed for use with Shugart 8-inch disks.

These 8-inch drives used the same soft-sectored data-storage format that had been used by IBM with 8-inch disks in the 3740 Data Entry System. You may at times see a reference to a "3740-compatible" floppy-disk drive; the term comes from this first use by IBM. (IBM has never made a 5-inch floppy-disk drive.) In the never-ending quest for increased storage capacity, many manufacturers have developed other formats for storing data on both 8-inch and 5-inch disks; most of these have used the more compact but less flexible hard-sector system.

What's All This Talk About "Winchester Technology"?

All of the ballyhoo about "Winchester" drives is the basis for disagreement as to the feasibility and practicality of a small rigid disk for personal computers. To understand why discussion is necessary, you must first know a little about the technology itself and how it came to be.

As mentioned previously, IBM first developed disk storage as an alternative to tape in the 1950s. Once a disk was full, data had to be "dumped" from the disk onto tape to create space for more data.

Although IBM continued to increase track and bit densities and access rate through media and head assembly improvements, it was clear that even more capacity was necessary.

In the early 1960s, IBM introduced replaceable rigid disks in the form of disk "packs." These packs contained as many as ten disks and, when mounted in the drive, required careful alignment with the head assemblies.

Meanwhile, improvements to the media and head assemblies continued to the point where only a thin layer of air separated each disk from its head. Contact between the head and the disk was catastrophic and was to be avoided. However, sometimes it did occur—either because the drive failed or the disk was contaminated. What happened as a result was called a "crash," an unfortunately appropriate description. Disks were damaged and, of course, data was lost.

Without further reduction in the flying height of the head (which would result in even more crashes), improvement in data capacity was not likely. In addition, the disks themselves—because of coating thickness, magnetic properties, and variations in disk smoothness—didn't allow for any further increase in track and bit densities. A totally new approach was required.

In 1973, IBM announced a new 14-inch disk drive, the Model 3340, an indication of what new thinking about an old problem could accomplish.

In concept, the new product was to be a dual-drive configuration. Each drive was to have a capacity of 30 megabytes, making the product a "30-30." As the story is told, because the caliber of the famous

James J Bochnowski
President
Shugart Associates

Disk-Drive Technology — At The Crossroads?

Microcomputer manufacturers are gearing up to accommodate the growing demand for personal computers. Sales in 1980 are anticipated by some to be three times those of 1979.

With this significantly expanded customer base, it is also expected that the cost of computer power will decrease. And, indeed, some manufacturers have already begun lowering prices.

But what about storage power? Are lower cost and increased performance in the floppy-disk drive's future as well? James J Bochnowski, President and Chief Executive Officer of Shugart Associates in Sunnyvale, California, believes that disk-drive technology—as it relates to personal computers—is nearing a crossroads.

"When the founders of Shugart developed the 5-inch floppy-disk drive in 1976, they expected to sell a total of 30 to 50 thousand units," says Bochnowski. "Last year, we sold more than half a million. Although we've made changes in the manufacturing process to lower production costs, we've gone about as far as we can. The 5-inch was never really targeted as the high-volume product it is, and its assembly costs reflect that.

"That's not to say we won't continue to make improvements in the drive itself, however," Bochnowski continued. "Our product plans include increasing access times and capacities for both the single- and double-sided 5-inch floppy drives. We'll accomplish this through an improved drive mechanism and better head assembly."

Bochnowski, who is obviously excited about personal computing, recently ordered a desk-top computer system for his home. He believes that microcomputer users will redefine disk-drive needs and dictate industry trends. "I think that, eventually, the personal computer market will be segmented in the same way that the small business systems market is," comments Bochnowski. "The spectrum of disk-drive products for personal computers will include low-cost, low-performance drives at one end, higher-cost, higher-performance drives at the other end, and moderately priced products in between."

What about 5-inch Winchester-type drives for personal computers? Bochnowski is not convinced that they're a cost-effective answer. "When we scaled down the 14-inch Winchester-type drive to 8 inches, we found that to uphold the Shugart philosophy and provide the lowest cost per function we had to make some performance trade-offs. To move down to 5 inches, more performance capabilities would have to be sacrificed simply to successfully utilize Winchester technology."

Bochnowski notes that the cost of a 5-inch hard-disk drive would not be much lower than that of the 8-inch. "Development

and special tooling economies have already been realized with the 8-inch Winchester-type drive, so no additional cost savings on the 5-inch unit are likely," he said. "And it's not clear yet that a 5-inch rigid drive would ever be a high-volume product, especially when you consider that the 8-inch product offers more capability at the same cost. However, Shugart has made no decision on this. We're open to discussion."

Is Shugart committed to the personal computer industry? "No doubt about it," claims Bochnowski. "The microcomputer market has, in a very short period of time, become a major application area for us. We keep close tabs on how end users are applying our drives, and the personal computer market is one of the top three applications for us. I might add that we believe it may well become the number one market for Shugart within the next few years."

Bochnowski also observes that, regardless of media and drive improvements that will be made in the coming months, significantly higher disk-storage capacity at substantially lower cost will require "totally new products with a fresh approach to disk-drive technology."

Asked if Shugart is exploring any new technological avenues, Bochnowski replied, "We're putting the resources—people, facilities, and equipment—in place now to meet the needs of personal computer manufacturers and end users both today and tomorrow.

"To sum it up," Bochnowski stated, "where there's a need, we'll do our damndest to fill it."■

Winchester rifle is 30-30, both the new drive and its technology were nicknamed "Winchester." And even though the dual-drive plans were abandoned, and capacity of the new, now single drive was increased to 70 megabytes, the appellation remained and became synonymous with this unique technology.

The primary differences between Winchester drives and earlier drives included moving the read/write heads and assemblies, as well as the disks, into a hermetically sealed disk-pack assembly, thus creating an enclosure within an enclosure. This was a radical departure from previous practices.

Also, the heads were designed to rest in a preassigned area on the surface of the disk when the disk was not spinning. No data is writ-

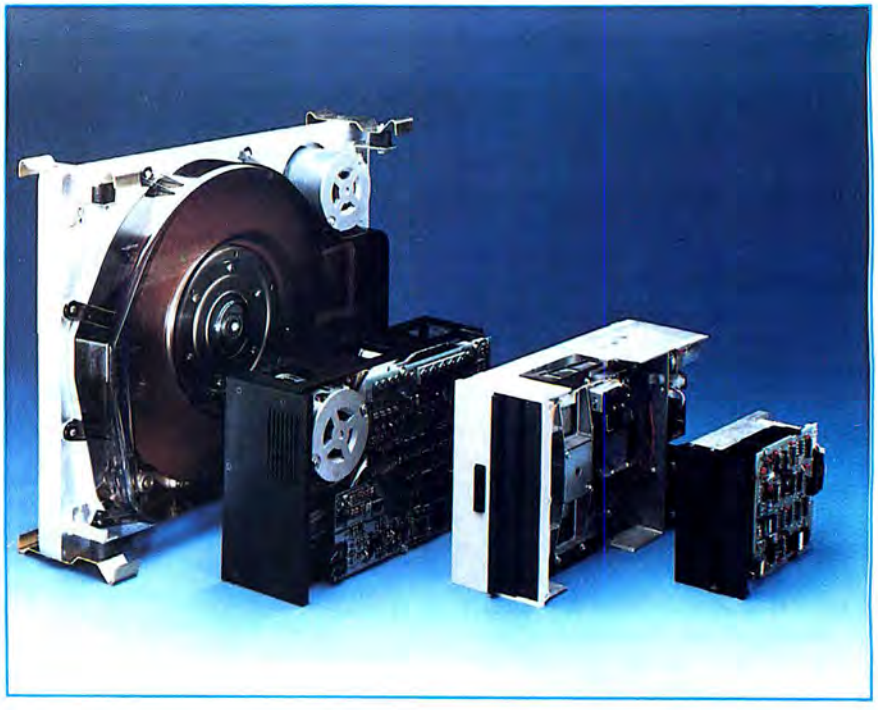


Photo 2: *Four different types of floppy-disk and hard-disk drives. Shown from left are: 14-inch fixed-disk drive; 8-inch fixed-disk drive; 8-inch floppy-disk drive (double-sided in this case — see text); and the 5-inch floppy-disk drive.*

ten in this landing area.

Aligning the heads and assemblies with the disks before they are sealed in the disk pack eliminated the need for critical alignment during installation of the drive. The motor, actuator, and control electronics remain outside of the sealed disk pack, added insurance against disk contamination through dust, heat, etc.

Of course, many other highly sophisticated technological improvements were made in addition to the ones mentioned. Thus, the cost of Winchester drives was prohibitive for minicomputer users.

When IBM announced the first 8-inch Winchester drive, it offered hope for smaller users. There were trade-offs, however. Scaling down the 14-inch Winchester to 8 inches meant either decreasing performance (track and bit densities and access time) to achieve lower cost, or maintaining the high performance and associated costs.

In the microcomputer industry, a 5-inch Winchester drive is being investigated. How much more performance would need to be sacrificed to bring the drive into an affordable price range? Since the approximate retail cost for an 8-inch Winchester drive is $2500+, is it even feasible to consider it?

How many personal computer users would appreciate the faster access, higher capacity, and increased data integrity enough to pay for it? Shouldn't double-sided floppy drives be pursued first? What about non-Winchester rigid-disk drives? Would they be practical? In general, would rigid disks be less expensive for the user in the long run than any type of floppy disk?

These and other questions are being asked by personal computer manufacturers. Only time and the demands of a growing customer base will provide the answers.

Summing It Up

As disk-drive technology evolves, enhancements that lower cost, expand capacity, and improve performance will encourage more people to discover the world of personal computing. It appears that, in the near future, floppy-disk storage will continue to be the primary choice of personal computer manufacturers and their customers for cost-effective data storage capability until the simplicity of operation and significant economic advantages of rigid-disk drives become as irresistibly attractive and affordable as the personal computer itself is today.■

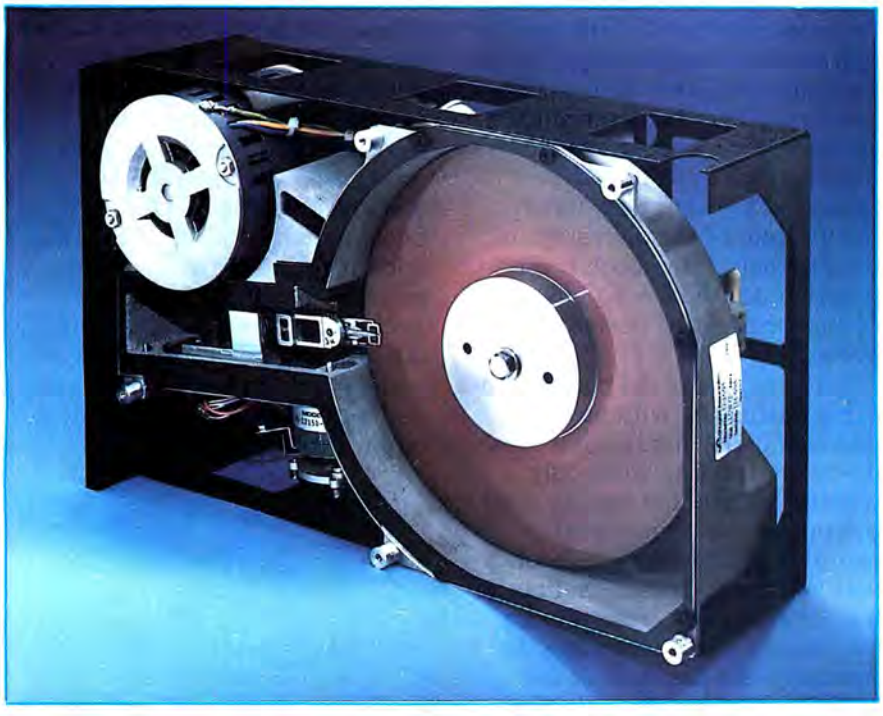


Photo 3: *8-inch floppy-disk drive with cover and printed-circuit board removed. Note the read-write head at left of center.*

All photos and figures in this article are reproduced courtesy of Shugart Associates.

Book Review

An Introduction to Microcomputers, Vol. 0: The Beginner's Book *by Osborne-McGraw Hill $9.50*
and
Understanding Microcomputers *by Scelbi Computer Consulting Inc $9.95*

Review by: John A Lehman

How does a beginner learn about microcomputers? A degree in digital engineering might solve the problem, but this isn't practical for most of us. To fill the need of the personal computer beginner for information, dozens of books have been published in the last few years. Some of these are excellent; some are terrible. I am going to describe two of the good ones—both published by respected companies. *An Introduction to Microcomputers, Volume 0, The Beginner's Book* is published by Osborne/McGraw-Hill. The other book is *Understanding Microcomputers,* published by the Scelbi Computer Consulting Company. Scelbi produced the first microcomputer kit, and was the first company to specialize in microcomputer software. One would expect these two companies to produce good beginner's books, and, indeed, they have done so. This review is a comparison of the two from multiple points of view: that of a fairly experienced microcomputer enthusiast and two beginners who read the books and commented on them for me.

Adam Osborne and Associates originally produced books on computer hardware, and this background shows in their Volume 0. The book presents the different parts of a microcomputer system, its development and use, the peripheral options, how binary numbers work, the internal working of the hardware, and how it is put together. The first half of the book presents personal computers through the experiences of Joe Bitburger, the archetypal beginner who, in the course of the book, buys a kit and builds a computer.

Most people who buy computers today buy them ready-made and working. A few years ago most people bought kits—kits that generally did not work when first assembled. All of the terrible things which happen to Joe in the book also happened to me, so I can testify that the description is accurate. While Joe is shown building and using an old-fashioned kit computer, ninety-five percent of the discussion in the first half of the book is appropriate for those who buy consumer-oriented personal computers.

The latter half of the book is concerned primarily with how microcomputer hardware works; if you are going to work only with BASIC and not modify your computer, you do not need to read this section, and the book tells you not to bother. Neither of the beginners who read this for me found it very interesting. In summary, the first half of the book is very good in explaining the hardware alternatives and what goes into a microcomputer. However, one complaint about the book is that it does not devote enough space to software and programming.

Scelbi Computer Consulting is basically a software company, so you might expect that they would have just the opposite bias. This is not really true. *Understanding Microcomputers* covers just about the same material as the Osborne/McGraw-Hill book, with less stress on nitty-gritty hardware and a little more space devoted to software and programming. I personally found the Scelbi book better balanced, but both of my beginners (neither with a hardware background) found the organization of the book somewhat confusing.

So which book should you buy? If I had to buy only one, I would choose the Scelbi book: not only because I feel it is better balanced, but with its larger type it is easier to read. I am not necessarily in the majority; the local computer store informs me that the Osborne/McGraw-Hill book is their best-seller with a wide margin. There is a simple solution, however: buy both! Each has information the other doesn't, and you'll probably understand the common information better if you read two different explanations of it.■

Yes! I sure do want to get into **Home Video** and I like your money-back guarantee.

Please sign me up for the next eight issues of Home Video—and bill me for $9.97. If the very first issue doesn't live up to my expectations, I may cancel my subscription immediately by returning my bill marked "cancel" —and if at any time during my subscription I become dissatisfied with the magazine, I may cancel and you will refund the cost of all remaining issues.

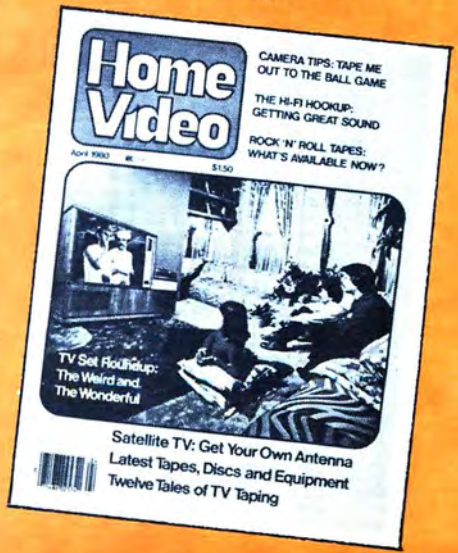
Name_____

Address_____

City_____

State _____ Zip _____

Enjoy the lifestyle of the eighties.



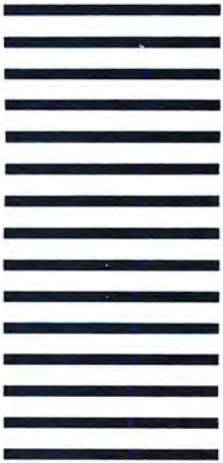
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 207 BOULDER, COLORADO

POSTAGE WILL BE PAID BY

Home Video

Subscription Service Department
P.O. Box 2651
Boulder, Colorado 80321



If you're into home video ...Now get into

Questions and Answers.
Every month, technical and non-technical problems solved by experts. Example: 20 questions answered on video discs.

Which Camera Should I Buy?
When you start making your own programs — a detailed report on the specs, prices, special features of more than 35 makes of color and B&W cameras on the market.

What's On Tonight?
— or today? Regular program guide to best network, PBS, Pay-TV shows for the month ahead.

Design for VCR Living.
Whether "VC" means "very comfortable" or "very chic," how to design and decorate around your new home entertainment center. In full color.

The Taping Game.
Interviews with VCR owners around the country, "just plain folks" and celebrities. What they tape, and why.

75 Sources for Videocassette Programs.
Complete listing of practically everything available, feature films, children's programs, lessons, X-rated, sports, classics, etc.

Tape Me Out to the Ballgame.
How network sports directors shoot a game. Rules for major league parks. Little League — make your own rules.

...the magazine that spearheads today's revolution in home entertainment, communications, lifestyle!

If you got a VCR for Christmas—or if you've subscribed to cable or pay TV—or if you've added projection TV in your game room—or if you've become a fan of video games—

—or if you're just on the verge of joining the more than 2,000,000 U.S. families who are already enjoying the fun and benefits of this great new communications technology—Home Video magazine is the one magazine that brings you all the news and information you need to get the most out of your equipment.

On these pages you'll see just a sampling of what readers are finding in Home Video (the first issue was a smash-hit on the newsstands) and what they have to look forward to in the exciting issues ahead.

Then, if you decide as hundreds of thousands of readers already have, that Home Video is for you, simply use the insert card—with a money-back guarantee if you're not completely delighted!

...Or use this full money-back guarantee coupon.

HOME VIDEO 50C6
P.O. Box 2651, Boulder, Colorado 80322

Please send me the next 8 issues of Home Video, for which I attach my check for $9.97. However, if at any time I am not completely delighted with the magazine, I may cancel my subscription, and you will refund the cost of all remaining issues.

Name

Address

City State Zip

Circle 50 on inquiry card.

onEducation

Computer Animation Helps Children to Learn

by Sam D Bryan

How do you capture the imagination of a child using a personal computer? Why not try animation!

I am the parent of a charming and exuberant young boy who, we have come to realize, is a "slow learner." My work as a computer systems analyst has brought me in contact with a variety of computer systems, and most recently with personal computers. I eventually bought one and began experimenting with educational programming.

For my first project, I decided to write a program that would teach a child how to match lowercase and uppercase letters. I started with a simple drill-and-practice strategy and a multiple-choice response format. Our son has a relatively short attention span, so I needed an engaging way to announce success and track progress. The program I eventually created uses a colorful visual aid: an empty "gas tank" displayed on the screen. Whenever the child gets a correct answer, some "gas" is added to the tank. The object of the game is to fill the tank as quickly as possible. The game itself consists of a series of four uppercase letters selected at random by the

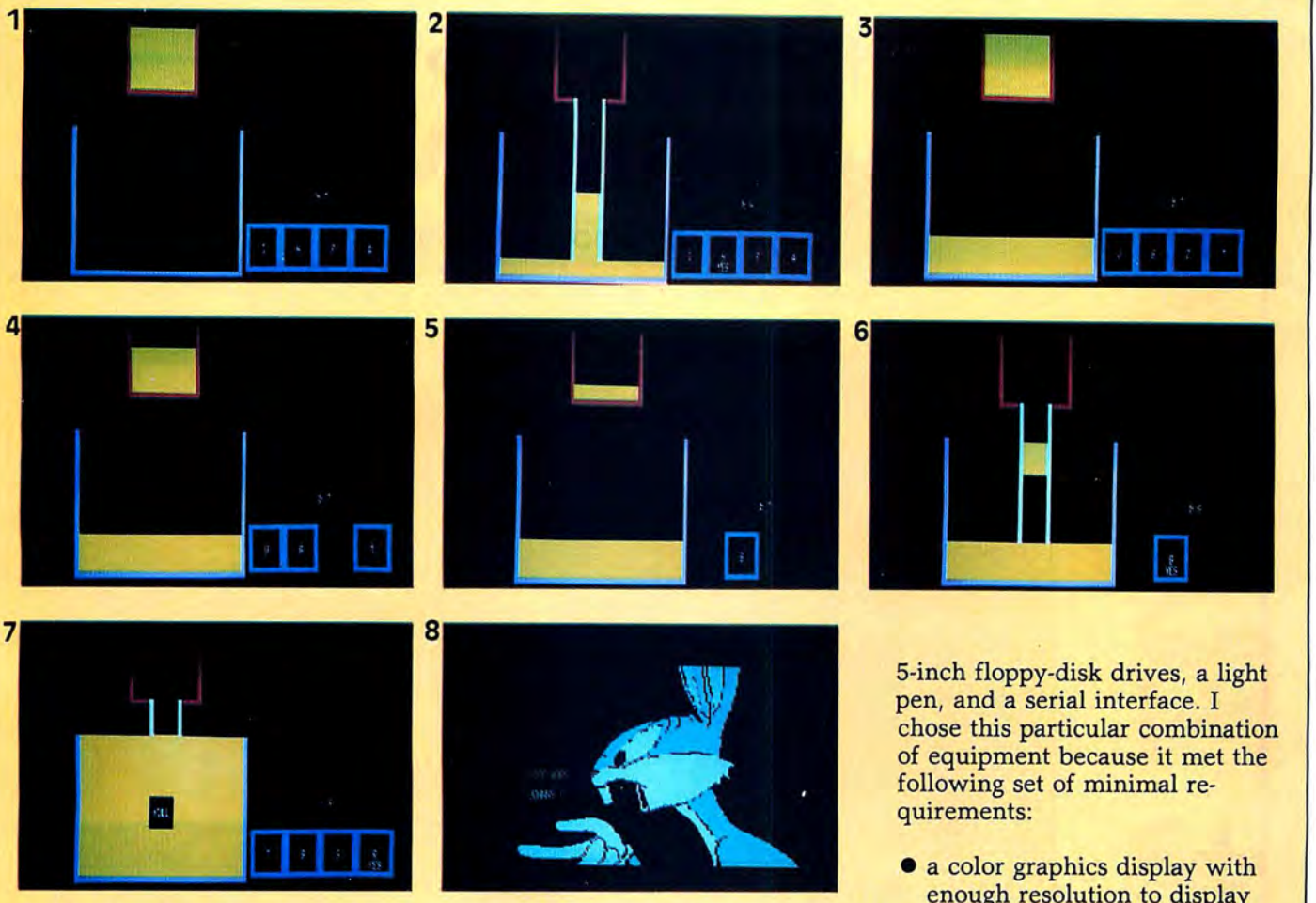


Photo 1: *Animation learning game. The bucket will empty into the tank whenever the student correctly matches the lowercase letter with the corresponding uppercase letter. The four possible choices are shown in the boxes at bottom right. It takes several correct answers to fill the tank. If the student answers incorrectly, one gallon of gas is removed from the bucket as a penalty, and the student is given another chance.*

Photos 2 thru 8: *Progressive steps in the tank-filling game. The student correctly guesses "H" in step 1, and the bucket empties into the tank. In step 2, "B" is chosen, which is incorrect. A gallon of gas is removed by the program. After two more incorrect guesses, the contents of the bucket go down again, and the student succeeds on the last attempt (thus the game stresses positive reinforcement). The last two frames show the tank being topped off. This is followed by a still frame of Warner Brothers' Bugs Bunny congratulating the student by name. (The Bugs Bunny graphic was designed by Kai Lanz of the Intecolor User's Group.)*

computer and presented to the child in boxes at the bottom right of the screen. One of these letters is then displayed in lowercase above the boxes. The child is then requested to choose which letter matches the lowercase letter. (In my particular setup, a light pen is used to simply touch the screen. The computer keyboard could just as easily be used.)

The presentation is designed to be fun to watch and to provide motivation to the student not only to get the right answer, but to finish the job by filling the tank. Incidentally, when the tank is filled, a still picture of Bugs Bunny appears on the screen along with a note of congratulations.

Hardware

The computer I used is the Intecolor 8052 personal computer, made by Intelligent Systems Corporation (ISC). It has a 19-inch, eight-color display monitor, dual 5-inch floppy-disk drives, a light pen, and a serial interface. I chose this particular combination of equipment because it met the following set of minimal requirements:

- a color graphics display with enough resolution to display cartoon figures
- a light pen available from the same company
- easy access to the graphics and light pen features with the existing software
- at least one year in the field and proven reliability
- ready repair available
- an active users group with a software clearinghouse
- an established company
- the computer can be used as a terminal

My program could also be written for the Radio Shack TRS-80, the PET, Apple, Atari, or other popular personal computer, and it could also be written for ISC's less expensive Compucolor II computer.

This small project was carried out for my own education, and while it only scratches the surface of computer-aided instruction, it achieved my immediate goals and taught my son about lowercase letters as well!■

Product Review

The APF Imagination Machine

by Robert Pieffer

Photos by Ed Crabtree

Photo 1: *The APF Imagination Machine. This $599 personal computer features 8 K bytes of programmable memory (expandable to 16 K), 10 K bytes of read-only memory, color graphics, built-in cassette recorder, a 6800 microprocessor, and video game module.*

Question: What has color like the Apple II, a built-in cassette recorder like the PET, a ROM-PAC slot and game paddles like the Atari 400, and a price like the TRS-80 Level I?

Answer: The APF Imagination Machine!

The APF Imagination Machine is an unheralded newcomer to the personal computer scene, but it has some interesting and surprising features for its $599 complete price. It is made by APF Electronics of New York City, an eleven-year-old company that started in the high fidelity business and later branched out into the calculator business.

My review machine arrived in good condition, and it worked immediately after I plugged it in. The photos reveal that the APF computer is actually a *hybrid computer*, combining a video game module (the APF 1000) and a separate computer keyboard module (the MPA 10). The two are connected by means of a U-shaped plug that is inserted between the slot normally

reserved for video game cartridges and a special slot on the keyboard module. Before you can insert this plug, however, you must first insert a rather awkward grounding strip inside one of the two slots. This procedure took me nearly 20 minutes; fortunately it must only be done once.

Features

The APF Imagination Machine uses the Motorola 6800 microprocessor and comes with 8 K bytes of programmable memory and 10 K bytes of read-only memory. The keyboard is standard, with fifty-three keys. There are two game paddles with twelve buttons plus a fire button and joystick each. The built-in cassette recorder is turned on and off by the computer. APF has announced a peripheral box for the computer that will enable the user to add an RS-232 serial interface so that printers and other peripherals can be attached. A floppy-disk drive made by MPI has also been announced.

The screen format is 32 characters by 16 lines, and the characters can be displayed in any of three different colors. Interestingly, there are three levels of resolution on the screen for graphics: in the first, or *base* resolution format, the user can display up to eight different colors with 64 by 32 individual color rectangles; in the second mode, called *high resolution graphics mode #1*, eight colors are available with a resolution of 128 by 192; the third mode, called *high resolution graphics #2*, features one color with a resolution of 256 by 192. The computer features a built-in sound synthesizer that plays music through your color television speaker.

BASIC

The BASIC interpreter for the APF computer is a fixed-point package, custom-engineered by APF. I would characterize it as being somewhere between the Radio Shack TRS-80 Level I and Level II BASIC interpreters. The upper two rows of keys on the keyboard, in addition to the usual characters, contain all of the BASIC commands such as PRINT, GOTO, and so on, and are accessible by holding down the control key and then pressing the desired key. This can save time when you're programming in BASIC. The BASIC package, lacking such things as sine and cosine functions, is primarily intended for business applications. However, the company is considering marketing a scientific BASIC package if there is a demand for it. I found the error messages to be particularly good with this BASIC interpreter.

Software

One clever feature of the APF Imagination Machine is an audio track on each program cassette. While you're waiting for the program to load, a prerecorded voice tells you about the features of the program you're about to use. And, of course, if you don't want to hear the sound track after the first few times, you can turn down the volume control. One caveat: I discovered that some of the program cassettes I received with the computer would not load properly. I called the toll-free hot-line number supplied with the machine's documentation and immediately got through to APF in New York. The people were very helpful and suggested that some of my tapes were from an early production run. The leaders on the tapes were too long and were being misinterpreted by the computer.

If you have this problem, try the following procedure: after typing lCLOAD, RETURN, RUN, and RETURN, the machine will display a message asking you to re-

Photo 2: *The two main sections of the APF Imagination Machine: the video game unit (at the left) and the computer keyboard module.*

wind the tape and hit PLAY. At this point, *wait approximately 2 to 3 seconds before hitting RETURN again*. The tape should load correctly. (Incidentally, the cassette recorder volume level is internally set.) The tapes loaded correctly every time after trying this trick. As I mentioned, the long leaders were in an early batch of tapes, and this problem has been corrected by the company. But there still might be a few left in some stores.

I received only four programs for review, but there is a sizable list of programs now available from the company. I particularly liked the Checkbook/Budget Manager program. It's well-suited to keeping a monthly budget. The Artist and Easel program was fun, although very slow to draw lines on the screen. The Music Composer is acceptable, considering that the circuitry probably consists of an 8-bit digital-to-analog converter (ie: a type of circuit that lets a digital computer drive an audio amplifier). Eight bits is somewhat low for accurate music reproduction, but at least the scale spans three octaves and is chromatic. I had some trouble getting the Typing Tutor program to work to my satisfaction.

Photo 3: *Installing the connector that links the two parts of the Imagination Machine.*

Of all the personal computers on the market, the APF unit's features probably compete most closely with the Atari 400

APF versus Atari 400

Of all the personal computers on the market, the APF unit's features probably compete most closely with the Atari 400, which costs approximately $500. How do the two stack up? Well, they both have low- and high-resolution color; the APF has a full keyboard, the Atari has a less elegant flat keyboard; the APF comes with a built-in cassette recorder, but the recorder is an optional extra with the Atari. One area where the Atari is going to be hard to beat is in software. There is a wealth of excellent Atari software available, including some impressive arcade-like games and educational games. However, APF seems to be making an effort to expand their software line.

Conclusions

Although the APF machine is not a well-known or established personal computer, it has some surprisingly good features, along with some quirks. Nonetheless, I suggest you give it an audition if you're looking for a computer in the $500 to $700 class.■

Photo 4: *Some APF software.*

THE SEARCH FOR A SMALL COMPUTER SYSTEM STARTS HERE

WE'VE TURNED MORE THAN 40,000 PEOPLE ON TO SMALL COMPUTERS—YOU'RE NEXT!

NATIONAL SMALL COMPUTER SHOW

October 30 - November 1, 1980

New York Coliseum

See more than 100 manufacturers of systems, peripherals, and software, and attend more thar thirty hours of free lectures. Don't wait on line in October, register now!

REGISTRATION FOR AMERICA'S BIGGEST SMALL COMPUTER SHOW

Please register me for the 4th Annual National Samll Computer Show, Oct. 30 - Nov. 1, 1980 New York Coliseum.

NAME _____ BUSINESS TITLE (If Any) _____

COMPANY (If Any) _____ TELEPHONE _____

ADDRESS _____

_____ ZIP _____

(Check main interest)

- ☐ Banking/Insurance
- ☐ Business office
- ☐ Communications
- ☐ Educational
- ☐ Government
- ☐ Hobby
- ☐ Industrial/Manufacturing
- ☐ Military
- ☐ Professional
- ☐ Stock Brokerage
- ☐ Transportation
- ☐ Utility
- ☐ Wholesale/Retail
- ☐ Other _____

(Check main job function)

- ☐ Accountant
- ☐ Administrator (Business)
- ☐ Architect/Builder
- ☐ Art Director
- ☐ Banker
- ☐ Computer technician
- ☐ Consultant
- ☐ Controller
- ☐ Engineer
- ☐ Industrial Designer
- ☐ Lawyer/law office mgr.
- ☐ Marketing manager
- ☐ Medical doctor
- ☐ Medical technician
- ☐ Office manager
- ☐ Programmer
- ☐ Public Servant
- ☐ Research/Development
- ☐ Teacher
- ☐ DP manager
- ☐ WP manager
- ☐ WP operator
- ☐ Stock Broker
- ☐ Systems Analyst
- ☐ Student
- ☐ Other _____

☐ ONE DAY $10 ☐ TWO DAYS $20
☐ THREE DAYS $30

Mail with payment of $10 for each day you wish to attend. Use one form per person. Registration badge will be sent by mail in early October. Check or money order only.

Mail prior to October 1, 1980.
National Small Computer Show,
110 Charlotte Place,
Englewood Cliffs, N.J. 07632
201-569-8542

Circle 60 on inquiry card.

Product Review

Photo 1: *The Atari 800, as set up in the onComputing offices. The computer is shown attached to a Texas Instruments video monitor, but it can also be connected to a color or black-and-white television set. The Model 820 printer and Model 810 floppy-disk drive are shown at left.*

The Atari 800 Personal Computer

by Ken Skier

Photos by Charles Frieberg

Atari, the company that put video games in millions of homes, has been marketing two new personal computers of late. The Atari 400 and Atari 800, with the generally advertised, respective prices of $549 and $999, are consumer computers, designed for people who may know little or nothing about electronics and programming.

Setting It Up

You open the box and it's all there: the hardware, the software, and the documentation. The power supply and cables are there, and they plug together in obvious, well-labeled ways. Furthermore, they go together only one way; you can't plug something in the wrong way.

The Atari 800 I received for review was up and running within five minutes after opening the box. As soon as it was set up I plugged in Basketball, a game cartridge that came with the computer. This game requires joysticks for each player's control.

There is no way a joystick plug can go into anything but one of the special trapezoidal joystick sockets.

Basketball

The game was great fun: I selected the "one-on-one" game and played until I was laughing too hard to continue. Little basketball players dribble, shoot, steal the ball from one another, block shots, and pass.

Those who already own an Atari video game may be wondering whether cartridges for your video game will run on the Atari computers. Unfortunately, cartridges for the video games are not compatible with the Atari computers. The only cartridges that will run on Atari computers are those specifically designed for them. But it seems reasonable to expect that most of the games already available on cartridge for the Atari video games will soon be available on cartridges for the Atari computers.

I cannot speak as a connoisseur of home video games, but I wouldn't be surprised if the Atari 400 and 800 are the ultimate home video games. Yet even if that's true, is it worth $500 or $1000 to have the best video game on the block? But they are more than video games. Each unit is, in fact, a computer designed as the heart of a computer *system*. So let's see how the Atari system stacks up, in each of three critical areas: hardware, software, and documentation.

Hardware

The hardware of the Atari 400 and 800 is a self-contained computer in a sealed case, including a full-size, typewriter-style keyboard; at least 8 K bytes of programmable memory for the 400 and 16 K bytes for the 800; circuitry and sockets that drive a television display (you supply the color or black-and-white TV) and let the computer "talk" to other devices such as joysticks, cassette recorders, printers, and floppy-disk drives; and one or two slots for program cartridges. Most of the differences between the 400 and 800 are in the hardware.

For this review, I had access only to an Atari 800, and therefore cannot comment on the Atari 400 from hands-on experience; but the Atari literature indicates that in many ways they are the same. The

most important differences lie in their keyboards and in their potential for expansion.

Keyboards

The Atari 400 has a one-piece, flat plastic keyboard. I imagine it's more like typing on a picture of a keyboard than on the keyboard of a manual or electric typewriter, because there could be little or no tactile feedback to each key-closure. The larger, full-stroke keyboard on the Atari 800 probably has a more comfortable feel, especially for a touch-typist like myself, than does the flat keyboard on the Atari 400. Otherwise, the two keyboards are similar: they have the same layout, and, with the same programs, will respond to the same keystrokes in the same way.

Television Display

The Atari 400 and 800 have the same video display circuitry,

Photo 2: *Atari software.*

which puts text and graphics onto a standard television screen. A cable coming from the Atari computer plugs into a switch box on the back of your television. The switch box, supplied with the Atari computer, is the same as the switch boxes supplied with other Atari products, and it connects in a minute or two. It lets you switch your television reception from the antenna to the computer-generated signal and back again.

The Atari computer produces a color television signal for channel 2 or 3, so you do need a color television to show off the Atari's display, though color isn't essential. A black-and-white set works fine: it displays colors as shades of gray, and this is suitable for most programs.

The display consists of 24 lines, with 40 characters per line. This character density is not outstanding: a number of personal computers, including the Radio Shack TRS-80, have 64-character lines. Some systems feature 80 or more characters per line, but such character densities usually require a video monitor rather than a television set, and few systems with such character densities offer color. Even so, a 40-character line is not likely to handicap you in any application except word processing, where it is desirable (but not necessary) to have a screen width as great as the width of the ultimate typed or printed page.

On the plus side, Atari gives you many ways to display each character. Each character on the screen may have any one of sixteen different colors. Any character may be displayed in normal mode (ie: light characters on a dark background), or in reverse mode (ie: dark characters on a light background). And you have 255 characters to choose from, including uppercase and lowercase letters of the alphabet, numbers, punctuation marks, and many special graphics characters. Thus, a wide range of characters and display options is available to the programmer and, ultimately, to the user.

Sound

The Atari computer drives not only the television screen but the television speaker as well. Computer-generated sounds can be directed to the television speaker, as can prerecorded voice or music as it is played back on the cassette recorder. Atari's Educational Programs exploit this feature by combining prerecorded cassettes with program cartridges to make the Atari a mixed-media teaching machine. Thus, a seemingly minor hardware feature, such as the capability to direct audio from a cassette to a television, can enable the development of a significant *software* feature. This may ultimately produce a richer experience for the user.

Joysticks

I love video arcade games because they're so detailed: cowboys draw and shoot; basketball players have all the real moves. But arcade games are also fun because, unlike conventional computer games, they don't require you to communicate via the keyboard. In an arcade game you fire torpedoes while looking through a periscope and pressing a red FIRE button; or, in an arcade gunfight, you draw, aim, and fire your gun by drawing, aiming, and firing a full-size pistol with a wire coming from it.

Thus, even if you can duplicate the display and sounds of an arcade game on your personal computer, you won't have the richness of an arcade experience if you have to communicate to the computer via the keyboard. Submarine commanders don't type out their

I played basketball with the computer until I was laughing too hard to continue playing.

orders— neither do gunfighters.

Likewise, the Atari doesn't require game players to type. Up to four joysticks may be plugged into an Atari 400 or 800. A joystick doesn't require any concentration or any training. With a joystick, even a first-time user can control a ping-pong paddle, a sports player, or virtually any other image on a video screen without having to look away from the screen or think about which keys to press.

Expansion

The Atari 400 and 800 are both very easy to expand, but an 800-based system can be expanded further than a 400-based system.

The Atari 800 has sockets on its side enabling you to plug in a number of peripheral devices (ie: accessories such as cassette recorders, printers, and up to four floppy-disk drives). The Atari 400 offers similar input/output sockets, but lacks the capability to connect to a disk drive.

In addition to peripheral units, another kind of system expansion involves plugging in more memory. There are two kinds of memory: programmable memory and read-only memory.

Programmable Memory

The Atari 800 can contain up to 48 K bytes of programmable memory, all of it internal; no external expansion unit is required. Apparently the Atari 400 cannot be expanded beyond the 8192 bytes of programmable user memory that come with the basic 400.

I expect that most consumers will buy the Atari 800 with the desired memory installed, or will have additional memory installed by the dealer. If you wish to add more memory on your own, it's a simple matter of getting the memory modules and plugging them in. Pictures in the owners manual show where the modules go for any desired memory configuration.

Program Cartridges

Read-only memory is even easier to add or replace than programmable memory. The Atari 400 contains one slot for a program cartridge, and the 800 contains two such slots. One program cartridge, about the size of a deck of playing cards, may be inserted into each of these slots, thus adding in an instant a large bank of read-only memory containing one or more ready-to-run programs.

Such program cartridges are not original with Atari. For example, the Exidy Sorcerer has had such "ROM-Pacs" for over a year now, but they're an excellent feature for any computer, and the Atari 800 is the first personal computer featuring slots for two such cartridges.

There are many advantages to program cartridges. First, they are *fast*. For most personal computers, a typical program or operating

Text continued on page 54

Nestar Lets Personal Computers Talk To Each Other

by Chris Morgan

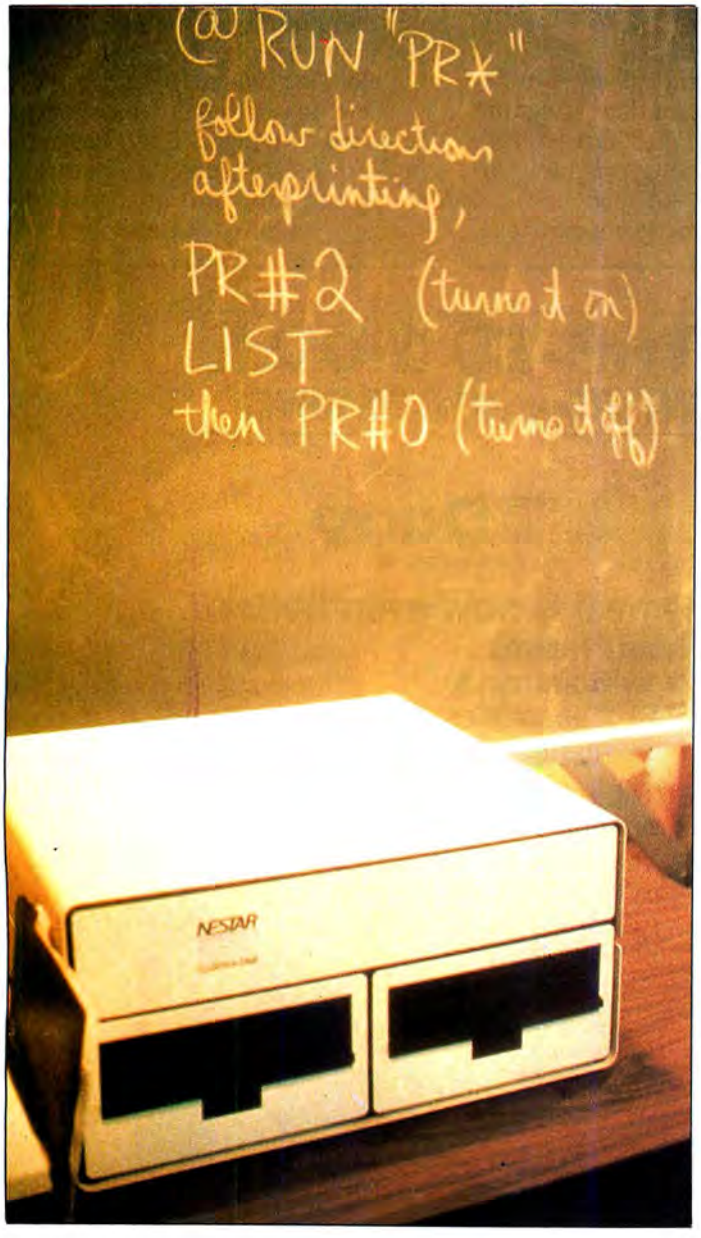
Personal computers will be a fixture in the classroom of the future, and Nestar Systems, a small company in Palo Alto, California, could have a big effect on the way they are used.

Nestar is located in California's "Silicon Valley," a dense pocket of computer and electronic companies south of San Francisco. I went to the Nestar offices recently to meet with President Harry Saal and see the Nestar system in action.

Next door to Harry's office the Nestar Cluster/One system was set up. It consisted of a master control box connected by cables to a dozen personal computers, such as PETs, Apples, TRS-80s, scattered around the room. Among other things, the Nestar system allows all of the students in the classroom to request the same program, if desired. It's a deceptively simple idea, the kind you probably thought had been implemented before. But the fact is that the Nestar system is the first of its kind.

Computer Headaches

At a recent symposium on personal computers in the classroom held at Harvard University, I heard teachers expressing some frustrations about using personal computers in their classes. A major problem was pointed out by the Lawrence Hall of Science's Arthur Luehrmann: with each student having his or her own cassette machine and cassette, it sometimes takes 15 minutes for all of the students in a classroom to enter their programs because of cassette loading errors, and so on. Very few schools can afford the luxury of equipping each student with his or her own floppy-disk drive — one way to speed things up. And there's the problem of lost or stolen student program cassettes.



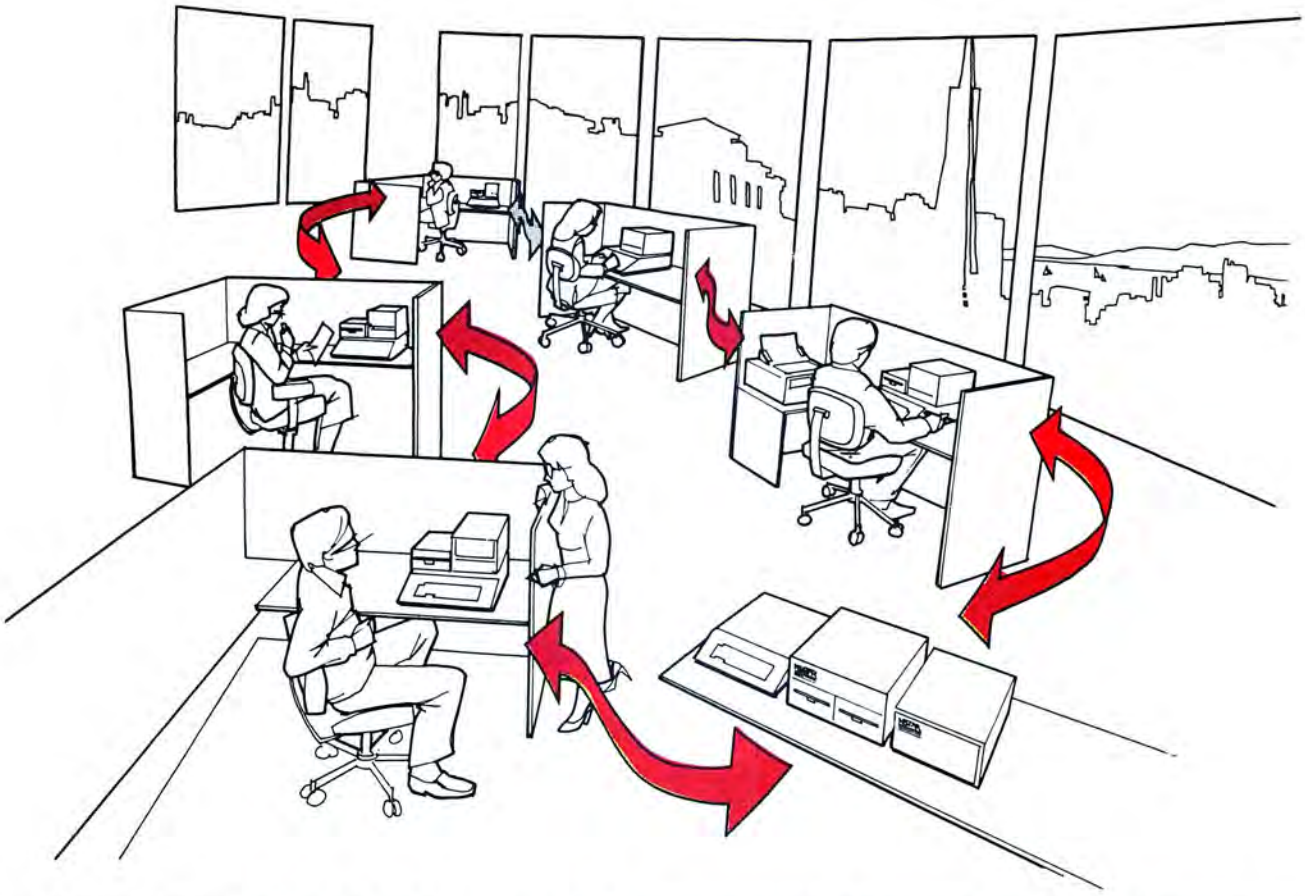


Figure 1: *Nestar's Clustershared Model A computer network, composed of up to sixty-four independent Apple II personal computers in a local communications, data, and resource-sharing network.*

A central system for disseminating programs is the obvious solution. Two such systems have recently come to light, manufactured by Radio Shack and Nestar. I'll let you know more about the Radio Shack system later.

The Nestar System

Nestar makes two different systems: the Cluster/One Model 1, their earlier system, and the new Cluster/One Model A. (The A is for Apple.) The original system allowed one to mix several brands of personal computers together on the Nestar *bus* (ie: a multiwire communications cable linking the network together). The system had limitations to the total number of computers that could be hooked together, and programs only, not data, could be sent back and forth.

The Cluster/One Model A is the newest Nestar system. It is a remarkably sophisticated network that allows up to sixty-four Apple computers to talk back and forth, exchange data, and share resources.

Using the Nestar System

Harry gave me a demonstration of the system. Imagine the following situation: your Apple computer is hooked up to the Nestar bus along with several other users. You can request a program from the central control module, which features two 8-inch floppy-disk drives and an optional hard-disk drive; you can send a program to be stored in the main memory and the program can be in BASIC, Pascal, assembly language, or whatever; you can request that something be printed out on a printer that is connected to *another* Apple computer on the system; and the list of possibilities goes on.

Resource Sharing

The Nestar system is an example of *resource sharing* on a local data network. It's not unlike *time-sharing*, but with several important differences. In time-sharing, several users share the same master computer. However, only one user can be serviced at a time; the busier the users get, the longer each may have to wait to get a job done. In the Nestar system, each computer maintains its own autonomy and its full computational ability at all times. However, all of the computers on the network can share, for example, a large, expensive line printer located at one of the stations. Only one computer can use the printer at a time, but each computer is free to do other tasks while waiting to use the printer.

Most of the time, the system is *transparent* to its users — that is, they are not conscious of its presence, because of its speed.

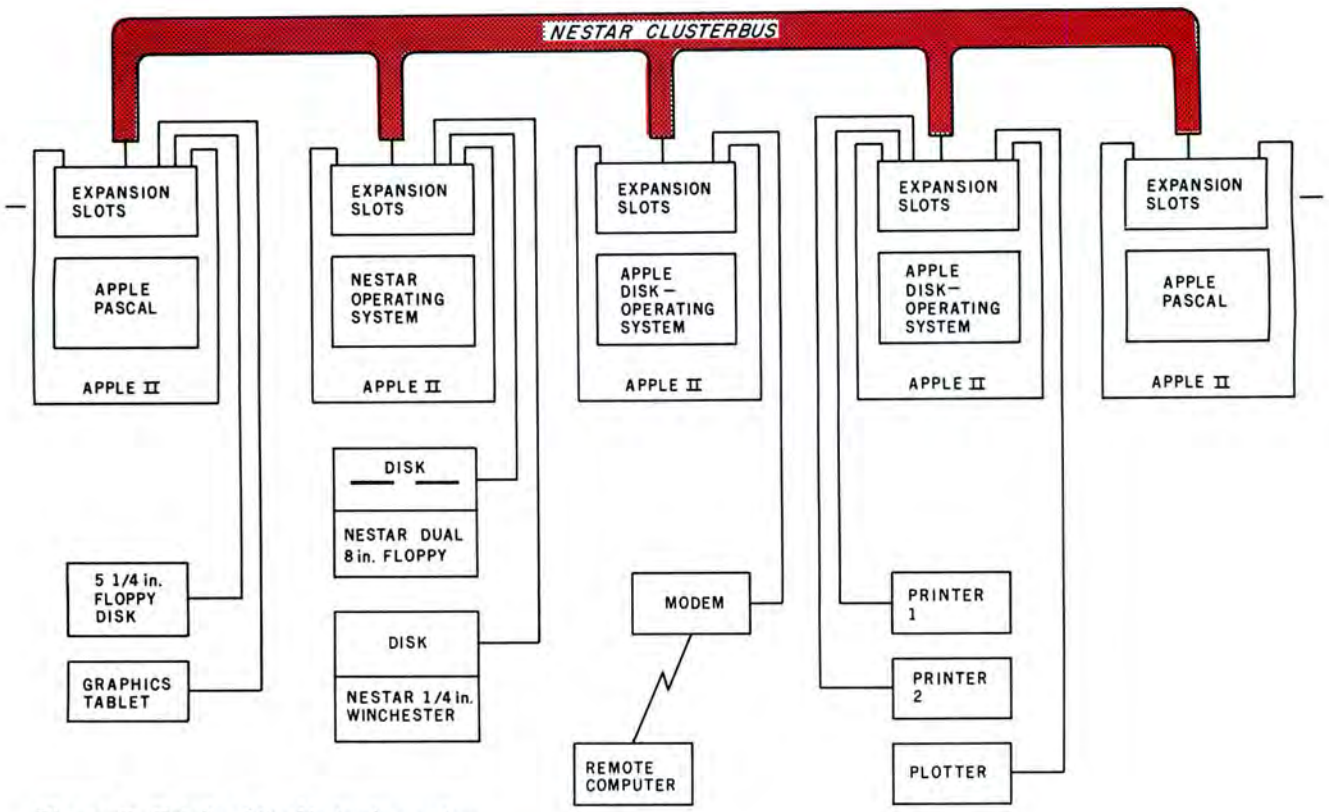


Figure 2: *Details of the Nestar bus system.*

Physical Setup

In order to hook up to the Nestar system, each Apple computer must be outfitted with a special printed-circuit card that plugs into the computer. This "buffered communications card," which costs $395, contains all of the necessary electronics to enable the computer to talk to all of the other computers on the bus, as well as the central module. A cable is connected to the card to complete the hookup procedure.

Harry Saal described some of the clever engineering built into the system: "One Apple computer in the system is used as the 'disk manager' — responding to data and program requests, directing 'mail,' and processing commands for each user station."

I asked what happens if several people want to read the same program from the central system at the same time. His answer was detailed: "If several people are either reading the same data or loading the same program, the system detects this and sends some or all of this information to be stored in the disk manager Apple. Each requester looks first in the disk manager computer for the desired program before tying up the main control-unit disks. So we have a one-level buffer that keeps the system from bogging down. In fact, the Nestar system is similar in concept to a new computer network plan from Xerox called Ethernet."

Additional Features

Any peripherals supported by the Apple computer can be used with the Nestar system, including speech-recognition devices, appliance-control devices, etc. And if any of the user computers fails, it doesn't affect the rest of the system. Users can secure their data files with password protection to prevent others from reading restricted information or changing programs.

Price

The Model A costs $6000 for the basic system with 1,260,000 bytes of storage. The optional 16.5- and 33-megabyte hard-disk systems cost $8000 and $10,000, respectively. And, as mentioned earlier, each station must be outfitted with a $395 communications card. For a typical ten-station system, the unit cost of the Nestar system (including all of the Apple computers) is about $2800. The price may seem high, but when you consider that it costs over $40,000 to educate a child from kindergarten through college today, this amounts to on-

Radio Shack's Network I Controller

The Network I Controller from Radio Shack is a $499 device that allows up to sixteen Radio Shack TRS-80 Level II computers to send and receive BASIC programs via a central 32 K-byte TRS-80 computer with at least one floppy-disk drive. (Therefore, it requires the expansion interface.) There are several constraints and limitations to the system in comparison to the Nestar system, but the prices of the two systems are significantly different.

In the case of the Network I Controller, the central TRS-80 computer system connects to it through the cassette interface. Sending a program to the central computer requires some synchronization of effort between the sender and receiver, preceded by the preliminary disabling of interrupts at both ends. The receiving system operator types CLOAD, then the sending operator types CSAVE. The program is transferred at the 500 bit per second (bps) cassette-data-transfer rate, considerably slower than the Nestar data-transfer rate of 250,000 bps. Following this, the receiving operator must save the program on disk. No other resource sharing is possible with this system. Nonetheless, it is a definite improvement over the individual loading of cassettes in a classroom.

ly a 7.5% increase in the total money spent.

Where Are We Headed?

We are only beginning to see the rich educational possibilities inherent in the personal computer. Sophisticated new hardware ideas like the Nestar system mean new hope for the future of computer literacy.■

Note: The Nestar system is being marketed through Apple dealers. For more information, contact Nestar Systems Inc, 430 Sherman Ave, Palo Alto CA 94306; telephone, (415) 327-0125.

Illustration by Robert Tinney

Living with Computers

by Jerry Pournelle

"Computers are great," said my friend, a famous author. "I've got to have one."

"Sure," I said, "but there are ways and ways to go about getting one . . ."

He didn't listen. And he's sorry now. A story goes with it . . . .

Whenever the Jet Propulsion Laboratories (JPL) hosts one of the solar system's great shows—Mars landings, encounters with Jupiter, whatever—there's usually a gathering of the science fiction clan, and we throw a party here at Chaos Manor. Of course I take every possible opportunity to show off my friend Ezekiel (who happens to be a Cromemco Z-2 computer with a lot of nifty extras); and the result is almost always the same. Everyone who sees him at work wants one.

The problems come when they decide to buy their own. Take four cases.

Case One is my writing partner Larry Niven. No fool he; when I first got Ezekiel, Larry came over and worked here. He liked what he saw, but he did nothing in haste. He waited until I had a system I was happy with; then he hired the same consultants (Proteus Engineering of Pasadena, California) to build him an exact copy, and bought the same software I had. The result was instant satisfaction. His system worked almost from the first day.

Almost: there were a few shakedown problems, mostly hardware, with one particularly plaguey failure cured only after spraying all his carpets with an anti-static goo because he kept zapping the encoder chip in his keyboard; but the important thing was the software. Even using the software I use he got in trouble a few times and almost lost some files—but because we both used the same stuff, I could tell him by phone how to cope. Incidentally, his result has been similar to my own, a near doubling of Larry's productivity and a sharp increase in quality of writing output—with the latter being by far

the most important. (See my comments on word processors in *A Writer Looks at Word Processors* in this issue.) The important thing to note is Larry's nearly painless transition to computers.

There's another approach, and it's disastrous.

Case Two involves my friend the famous author—he really is famous, and I'm withholding the name because you'd recognize it and I wouldn't want to embarrass him. Thinking that Ezekiel was marvelous, he couldn't wait to get his own. His problem is that he has too much money. When he got home he fell into the hands of consulting engineers who told him all of the great things computers could do, marvels far beyond Ezekiel's more modest capabilities, and he believed every breathless word of it. He called me to tell me what my problems were.

"You didn't spend enough money," said he. "I've been studying what these machines can do, and just wait until you see what I'm getting."

He was so enthusiastic that he went out and got a contract to write a book on microcomputers. I haven't seen the manuscript, but his outline said he would tell how the machines work, what they can do, how they'll change your life, how his microcomputer changed his life . . . .

Only one trouble. A year went by. His book was due. And the engineers were still telling him how great his system would be when he finally got it. He ended up writing his book on a typewriter, and as far as I know his system isn't working *yet*.

Now that's an extreme case, and most people shopping for a microcomputer aren't rich enough to generate themselves that kind of problem, but it's illustrative all the same.

Case Three is intermediate. It involves another famous author (Gordon Dickson) who confessed he knew nothing about computers, but liked what mine did. If he'd lived in California, he'd probably have done as Larry Niven in Case One did and simply bought a copy of Ezekiel, but given that he's a good thousand miles away that wasn't possible. As the next best thing, his consultant spent some time on the telephone with Tony Pietsch (owner of Proteus) before buying the system.

Unfortunately the two engineers didn't discuss the right subjects. Engineers never do. What's important to them is not necessarily what's important to the user. The result was a system that works, and works well. Gordon Dickson will never go back to a typewriter after using a word-processor program. But what he has is a system more adapted to a programmer than to a creative writer, and although he's only working about a third as hard as he did with his Selectric II typewriter, he's still working twice as hard as he would with the proper system.

Case Four is much like the third. Screenwriter David Gerrold, who used to work with a magnetic-tape Selectric, decided to get a full computer. Being a writer and not an engineer, he wanted something off the shelf. Not wanting to sink loads of money into a system, he got only the essentials. The result is similar to Dickson: he loves the system, but he's working harder than he ought to.

Neither Dickson nor Gerrold are in unrecoverable situations. They bought machines that can be upgraded to full capability. That makes them a lot better off than some writers I know who got nonstandard equipment and soon ran up against inherent limits that can't be overcome without scrapping everything and starting over. (I could tell a number of horror stories like these.) It's a trap that's all too easy to fall into, and worth a lot to avoid.

Which brings me to the point of this article: if you decide to live with a computer, how do you go about choosing one?

The Micro Is an Attractive Mistress . . .

First, please understand something very clearly: for most people—certainly just about all of the professional writers I know—buying a computer is not at all similar to buying any other durable good. It's more like getting married. You will find that you spend a lot more time with your computer than ever you did with a typewriter. There's something attractive about the machines; as Gordon Dickson says, a typewriter and blank paper tend to force you away, to make you think of other things that need doing, like sharpening pencils and carrying out the garbage. The computer, on the other hand, draws you into it.

Not that you'll spend all that much more time *writing*. Again with few exceptions, most writers who have acquired machines probably spend less time actually writing than they did before they bought them. They turn out more copy because the machine makes it so much easier to write. They

turn out *better* copy because a good text editor lets you endlessly rewrite without pain. You can change words and lines without retyping a whole page. You find yourself doing minor touchups here and there that you'd never do on a typescript; playing with words, trying this phrase and that to see how it looks before making a final choice—and no choice is really final until you've mailed the manuscript anyway. So yes, computers let you finish more work and get it better, and do it in less time, because they take most of the mechanical work—which is half the pain—out of writing.

But they don't give you more time, because the machines themselves are fascinating, and they eat up all the time they've saved and more. They're fun, and they're hard to resist playing with.

Marriages Aren't Made in the First Computer Store

You wouldn't marry the first woman you saw just because she looked good in a sweater (for the women reading this, change sexes and adornments appropriately). Similarly, you shouldn't buy the first microcomputer you see demonstrated. I know the temptation is great, because I very nearly did it. But don't. Give yourself a few days. A few weeks, even, if you can wait, although I know that's asking a lot from someone in love. But please, try to be rational.

Rationality consists of getting some idea of what you want a computer to do. It's not easy, because few beginners have any idea of what these machines *can* do; the analogy with marriage is a good one. Until you've tried it you don't know what it's like; until you've been happily married you don't know what it *can* be like. And until you've lived happily with a computer, you can't even guess how they can change your life.

Computers let you finish more work and get it better, and do it in less time, because they take most of the mechanical work — which is half the pain — out of writing.

So the first rule is, look at a number of systems. Visit friends who have one. Join a local computer club. Spend a bit of time in a reputable computer store; if possible, visit *several* computer stores, even if that means a weekend trip to a big city. (Take your family along and make a holiday of it. They'll see little enough of you after your system is installed.)

Above all, read the better computer magazines. This one, of course, and its sibling, *BYTE*. Get *An Introduction to Microcomputers, Volume 0: The Beginner's Book*, by Osborne/McGraw-Hill books, and read it.

When you've done all that, you will still know no more about living with a computer than marriage manuals can tell you about married life, but you'll know a few questions to ask when the marriage brokers—uh, excuse me, computer-store salespeople—try to sell you a machine.

What Do You Expect, Anyway?

Rule One: Whatever you expected from your computer, the good ones will be able to do more.

Rule Two: However much you get from your computer, it won't be quite enough.

On these two rules hang all the leisure and the profits.

Who Is This Guy to Tell Me About My Future Wife [Husband]?

My qualifications to give pontifical opinions: First, I'm a professional writer. I work at home, and for the past two years I've spent an average of eight hours a day with Ezekiel. Nowhere near all that time is consumed with creative writing. When I bought my system I was determined to remain a user and not get involved with programming and hardware, but I'm temperamentally unable to do that, as I suppose I ought to have known; you don't become a science fact/fiction writer unless you're driven by insatiable curiosity. Thus I read all the magazines, I eagerly play about with other people's systems, I go to computer shows, I own nearly every programming language capable of being run on the Z80 processor, and I've written a lot of programs that work.

Of necessity I run a small business. Most writers neglect the business details of their profession, and thereby lose a lot of money; if you put too much time into business you don't write, and if you don't write there's no business to manage. Fortunately I've found that Ezekiel can take care of an awful lot of the detail work, letting me do both. Meanwhile I've become familiar with a great number of business-oriented programs, and I know something of the computer requirements of small businesses: keeping books and records, keeping track of con-

tracts, generating contract changes, keeping files of books in print, making sense of data files, answering correspondence, generating the myriad of required government reports, and in general handling the time-consuming chores.

I also know what most computer enthusiasts find out too late: the machines *can* do all that, but most of them *won't* because there's no software; and even if there is software, it seems to be a law of nature that computer hackers can't write English, so you'll never figure out how to *use* the software that does exist without becoming something of a hacker yourself.

So: with all that experience behind me I know something of how to live with computers, and I've become a maniac on the subject. I love them. I can't conceive of life without one, and when my machine is down—fortunately a very rare situation—I feel totally lost.

Thus this article. Not everyone who reads onComputing is going to become a true computer nut, but the percentage of those who buy machines "just to use" and then find the darned things taking over their lives is larger than you'd suppose. Be warned.

Random Personal Recommendations

Up to now I've kept it general; but I was taught in the only writing class I ever took, Intro to Journalism, that one ought to be specific wherever possible. The trouble with recommending specific systems is that it's a bit like telling someone whom to marry; as long as you keep the advice general you're fairly safe, but when you get down to names you can have trouble. Similarly, if I recommend specific systems I'm going to be accused of slighting some just as good as those I like—and some of the accusations may be well-founded.

The following, then, are personal opinions based on over two years of experience with microcomputers. I don't think someone buying a computer would do well to ignore them, but they are not gospel either.

(1) Since in my experience your computer is going to absorb far more time than you think it will, and will be asked to do far more than you ever thought it would, it's best to get as versatile a system as possible; and in the Z80 field that means a system with an S-100 bus, using 8-inch soft-sectored floppy disks, and running under the CP/M operating system. There's as much software for that configuration as for the next three put together.

(2) Get as large a box as will fit where you intend to keep the machine. Be sure there are plenty of empty slots for circuit boards. There are an amazing number of special-purpose boards, including clocks, telephone answering devices, music boards, speech synthesizers, and gizmos to let your microcomputer control your house; unless there's room in your system you won't be able to add bells and whistles. My own system resides in the Cromemco Z-2 "black brick" box, which is built to withstand small atomic weapons, and I'm very satisfied with it. Doubtless there are many others as good.

(3) It's worth something to have the shop install a larger fan in the box. Over 75% of the hardware problems I've seen in microcomputers have been heat dependent.

(4) If you've got to save money now and you're not in a business that demands pretty print, save on the hard-copy device and plan to replace it later; don't stint on the basic electronics.

(5) Get one of the Big Three text editors (Electric Pencil, Magic Wand, or Word Star) and get it operating immediately. Also acquire Word Master; you probably won't use it as a primary editor, but when you do decide to do programming there's nothing to equal it.

(6) Right now do something nice for your family. They're going to miss you after your computer moves in . . . .■

About the Author

Dr Jerry Pournelle has for the past nine years been a full-time writer of science fact and fiction. He is the former director of the Human Factors Laboratories for the Boeing Company, and was involved with projects Mercury, Gemini, and Apollo, as well as military aerospace systems.

Dr Pournelle holds degrees in engineering, psychology, and political science, and was the successor to the late Willy Ley as science editor for Galaxy Science Fiction *magazine, where his column was a popular feature. At present he is a regular columnist for Ace Books'* Destinies *magazine and* Analog Science Fiction, *and will shortly begin computer equipment and software reviews for* BYTE *magazine.*

With Larry Niven he is coauthor of the best-selling novels Lucifer's Hammer *and* The Mote in God's Eye. *Winner of the first John W Campbell memorial award, he has written a dozen novels of his own. His latest novel is* Janissaries *(Ace Books), and his latest nonfiction work is* A Step Farther Out *(Ace/Grossett).*

Everybody's making money selling microcomputers. Somebody's going to make money servicing them.

New NRI Home Study Course Shows You How to Make Money Servicing, Repairing, and Programming Personal and Small Business Computers

Seems like every time you turn around, somebody comes along with a new computer for home or business use. And they're being gobbled up to handle things like payrolls, billing, inventory, and other jobs for businesses of every size...to perform household functions like budgeting, environmental systems control, indexing recipes, and more.

Growing Demand for Computer Technicians... Learn in Your Spare Time

Even before the microprocessor burst upon the scene, the U.S. Department of Labor forecast over a 100% increase in job openings for the decade through 1985. Most of them *new* jobs created by the expanding world of the computer. NRI can train you at home to service both microcomputers and their big brothers. Train you at your convenience, with clearly written "bite-size" lessons that you do evenings or weekends without quitting your present job.

Assemble Your Own Microcomputer

NRI training includes practical experience. You start with meaningful experiments building and studying circuits on the NRI Discovery Lab.® Then you build your own test instruments like a transistorized volt-ohm meter, CMOS digital frequency counter...equipment you learn on, use later in your work.

And you build your own microcomputer, the only one designed for learning. It looks and operates like the finest of its kind, actually does more than many commercial units. But NRI engineers have designed components and planned assembly so it demonstrates important principles, gives you working experience in detecting and correcting problems. It's the kind of "hands-on" training you need to repair and service units now on the market.

Mail Coupon for Free Catalog No Salesman Will Call

Send today for our 100-page, full-color catalog. It describes NRI's new Microcomputer Technology course in detail, shows all equipment, kits, and lesson plans. And it also tells about other NRI courses... Complete Communications with 2-meter transceiver...TV/Audio/Video Systems Servicing with training on the only designed-for-learning 25" diagonal color TV with state-of-the-art computer programming. With more than a million students since 1914, NRI knows how to give you the most in home training for new opportunity. If coupon has been removed, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

RUSH FOR FREE CATALOG

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

Please check for one free catalog only.
NO SALESMAN WILL CALL

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Amateur Radio • Basic and Advanced
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating Including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 177-050

Word Processors
A Look at Four Popular Programs

by Larry Press
Small Systems Group

Word processors let you edit and format text quickly and efficiently using a personal computer.

Word processing—what is it, and what can it do for you? Today's personal computer media are filled with word-processing advertising. It seems that every package has its own set of special features, and the beginner may have some trouble in sorting things out.

To help you get started, I will compare four word-processing packages: Auto Scribe, Electric Pencil, Magic Wand, and Word Star. With the exception of Auto Scribe, all of the programs reviewed in this article are CP/M versions. The Auto Scribe package we evaluated is designed for the North Star Horizon computer. CP/M is a popular disk operating system for use with 8080- and Z80-based computer systems. (Electric Pencil is also available for the Radio Shack TRS-80 personal computer.) But before turning to these specific programs, we need to consider a few background factors—the types of programs available, the ways in which a word-processing system can be used, and some hardware considerations. Once the stage is set, I will review the programs.

Types of Programs

There are three types of programs used for processing text: editors, print formatters, and word processors. An editor is a program used to create and modify text for subsequent printing using a separate program called a print formatter. For example, in preparing an article I would begin by keying in the text using an editor. Once it is entered and corrected, the session would be terminated and the resulting file saved on the disk. I could then use a print formatter to read the saved file and prepare the final printed output. If I wished to make subsequent changes, I would use the editor again to modify the original file, and produce an updated version for reprinting. (See figure 1.)

A word-processing system provides the editing and printing functions in one package. Some companies choose to do this with one integrated program. Others provide a separate editor and formatter designed to work well together. You may have heard a distinction made between *line* editors and *screen*, or *context*, editors. Line editors are generally designed to work on printing terminals, and are not as well suited to word processing. All of the systems reviewed in this article use screen editors.

Screen editors assume that you have a video display which lets you see a portion of the text in memory. In a sense, the video screen is a *window* that can be moved around on the text. The user moves or *scrolls* the window to the appropriate position in the text file and makes changes by retyping what appears.

This is an intuitive and direct way to work. You don't spend time checking to see where you are, checking to see what you've done, and composing commands, as you have to do when using a line editor. You're free to concentrate on composing text, so learning to use a screen editor is relatively easy.

Despite their limitations, a number of line editors have features which are useful for composing rigidly formatted files which consist of separate lines; for example, computer programs. It is also possible with some line editors, if you are an advanced user, to write programs of command sequences, enabling you to do some tricky things.

There are also two general philosophies regarding print formatters. At one pole is a program in which you format the document

Word Proceszors

Word Proceszors

Word Proceszors

Word Proceszors

Word Proceszors

Word Processors

Word Processors

Word Processors

Word Processors

Word Processors

Word Processors

being worked on so that it appears on the screen exactly as it would be printed. Auto Scribe and Word Star lean in this direction. At the other end of the spectrum, the appearance of the text on the screen bears no relation to the way it will appear when printed, and various commands are embedded in the text to control print formatting. Electric Pencil and Magic Wand are more reliant upon embedded commands than the other two packages; however, all systems fall somewhere between these poles.

The first extreme—what you see on the screen is what you get on the printer—sounds great. It would simplify your job: you would never be surprised by any quirks in your printout since you would have previewed it on the screen, and there would be no print formatter commands to learn. Unfortunately, there are some problems that keep us from achieving this goal.

For one thing, the hardware which is commercially available at reasonable prices is not up to the task. In general, video screens are not capable of displaying material as it will eventually be printed (unless we severely limit what we can do with printers). For example, my high-class printer is capable of superscripting and subscripting, of printing lines which are 160 characters wide, and of varying the widths of printed characters, the spaces between characters, and the spaces between lines. My video display cannot do any of this, so it really cannot show me exactly what is going to be printed out. And in many applications, the material to be printed is supplied at the time the document is being printed, not while it is being composed. The operator might be supplying amounts owed in the case of an accounts-receivable dunning letter, or names and addresses may be read in from a data file on the disk at the time a letter is being printed.

A print formatter that allows for embedding commands can also be made to do some sophisticated decision making at print time by allowing the experienced user to essentially write small programs of command sequences. Finally, it takes some processing time and perhaps operator action to reformat text so that it appears as it would be printed. Given these sorts of factors, all systems allow for some embedding of print commands in the text, although they vary in degree.

There are editors and print formatters, but what about full word-processing systems? With a word-processing system you're able to create and modify files (ie: to edit them) and to print them out. This may be achieved either by having you compose your document, then save it on the disk for printing (as with Magic Wand or WordStar); or it may be possible to print a document while it is in the process of being edited (as with Auto Scribe or Electric Pencil).

Who Are You?

The first step in shopping for word-processing software is to ask yourself what sorts of things you plan to do with your system. A small business which sends out much repetitive mail will look for different things than a journalist who uses a word processor to write articles. Let's look at some people who might use a word-processing system.

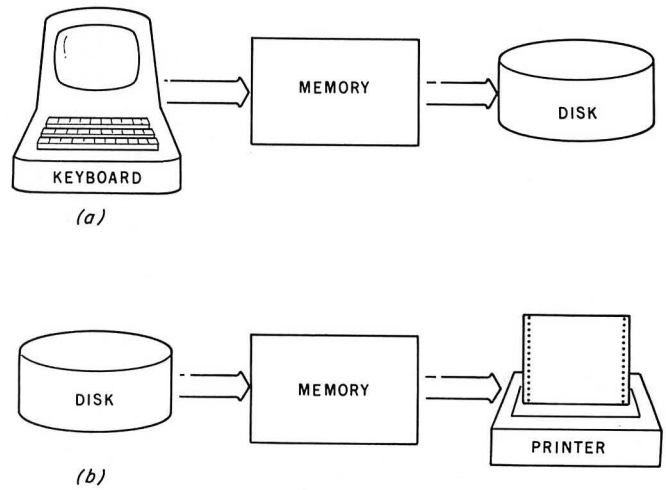


Figure 1: *The two parts of a word processor. Shown at (a) is the editor, which lets the user enter text, revise it, and save it on the disk for further revision or for printing. At (b) is the formatter, the program which controls how the printed text will look. With a formatter, the user can set margins, justify text (ie: line up the ends of text lines), and so on.*

There are three types of programs used for processing text: editors, formatters, and word processors.

An Author

The author is going to use a system to write and revise relatively lengthy, unique documents such as articles, stories, and proposals. Jerry Pournelle discusses word processing from this point of view in his article on page 83 of this issue of onComputing. An author keys in a large quantity of text and makes repeated revisions, so there is great interest in editing features. It is advantageous to be able to move the cursor quickly around on the screen and to have a system that scrolls the text rapidly. Deletions and insertions must be simple and quick.

Print formatter features, such as the ability to read names and addresses from a disk file at the time a document is being printed or being able to vary the printout depending upon what an operator keys in response to a fancy prompt, are not important to the author. On the other hand, some print-time features such as automatic page numbering, headings, and footnotes are important.

A Marketing Manager

The marketing manager of a small business will use his or her word processor in different ways than an author. He or she may wish to send a new product announcement to 150 magazine editors or send a "personalized" letter to 3000 past customers. Since the announcement or letter is fairly short and a new one is composed infrequently, the manager can live with an editor which is a little slower and more confusing to use. On the other hand, the sorts of print formatting features which are of little interest to an author are critical to the marketing manager.

This formatter should be able to accept inputs like names and addresses from customer files on the disk or from the operator while it is printing. Features which give a degree of programming flexibility are important in cases such as this. For example, the marketing manager's system may be used to generate different letters for different sorts of people in the customer file, so the print formatter must be able to read information into variables and then test their values. If the word processor is used to prepare and maintain mailing lists and other data files, there will be interest in having a system that can prompt the operator and which can write to the disk, as well as to the printer.

A Secretary

A secretary falls somewhere between the writer and the manager. In doing general correspondence, many unique documents will be composed; however, they will be relatively short and will not be subject to much revision. Therefore, editing features will be somewhat less important to a secretary, and it is okay if the system slows down when editing long documents.

While each letter is unique, each has a relatively similar format (date, address, salutation, body, etc). So, features that automatically position various parts of a letter are useful. Being able to display a close approximation of what will eventually be printed is also handy for this type of work, as is the abili-

Word-Processing Programs Reviewed

Program	Manufacturer	Price
Auto Scribe	Micro Source 1425 W 12th Pl Tempe AZ 85281	$400 (suggested retail)
Electric Pencil*	Michael Shrayer Software 1198 Los Robles Dr Palm Springs CA 92262	$250 thru $300 (depending on the printer used)
Magic Wand	Small Business Applications 3220 Louisiana St Suite 210 Houston TX 77006	$400 (approximately)
WordStar	Micro Pro International 1299 4th St San Rafael CA 94901	$495

*A new character-oriented version of Electric Pencil, called Electric Pencil II, is available for the Radio Shack TRS-80 Model II (and other CP/M systems having the DMA feature) for between $275 and $300. It was not available in time to be included in this review. . . ed

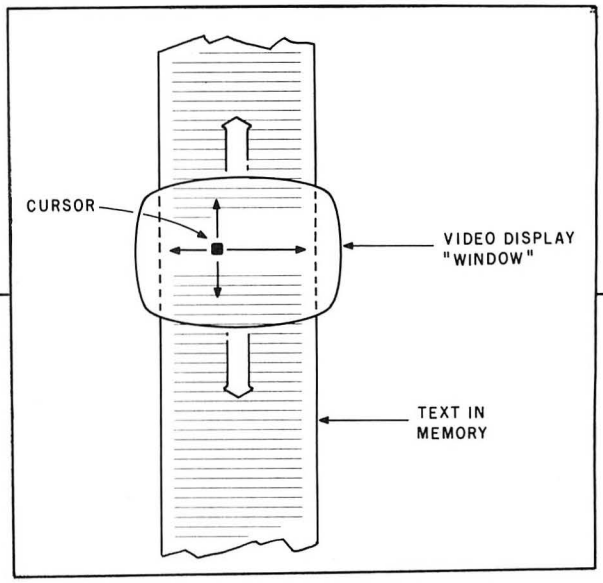


Figure 2: *Viewing the text. The video display acts as a text window into memory, allowing the user to see a portion of the text at a time. Within this area, the cursor can be moved around to add or delete information.*

ty to print copy without saving a file on the disk or leaving the editor to run a separate formatter.

A Newsletter Composer

A word processor may be used to prepare camera-ready material for printing. In this case, the documents will be fairly large and unique. Like an author, a composer will be interested in editing features during input and while correcting typographical errors, but won't spend as much time on revision. On the other hand, there will be more interest in print formatting features.

In composition, the ability to utilize the special features of the system printer is critical. For example, proportional character widths (ie: wider spaces between wide letters than narrow ones), precise control over the distance between successive lines, boldface printing, superscripting and subscripting, and justification (ie: lining up the left or right hand edges of the text, as in a newspaper text column) are needed in this case. In addition, it is convenient for a composer to have control over formatting at the time of printing, in case some trial and error is necessary. For instance, if an article is printed and it is discovered that it is slightly too long as formatted, the composer might want to reprint it with 1/48 of an inch less space between each line. Again, a composer would like to see a good approximation of the way the material will eventually be printed while it is being edited.

After comparing the programs under review, I will make some specific recommendations for each of these users. First, however, let's look at hardware considerations for word processing.

Hardware Considerations

The toughest decision in putting together a word-processing system is choosing software; however, there are a number of hardware choices which must also be made. How much memory do you need? What size disk drive? What sort of printer? Should you get a terminal or a direct memory access (DMA) video display? (We'll define DMA later in the article.) Let's look at these questions.

Memory

The question of how much memory to get is easily answered: get 64 K bytes unless you absolutely cannot afford it. If the documents you work on are all short enough to fit in your computer's memory (along with the program), your word-processing system will be fast. However, as soon as your document length becomes greater than the size of memory, things slow down because the system must begin saving parts of the material on the disk. This may be done automatically or may require operator action. In either case, it will be slow and bothersome. The other obvious bonus to getting the maximum amount of memory is that other tasks you may want to do with your computer can probably be accomplished more quickly and easily.

Disk

With floppy and hard disks you have a wide range of choices. (See "Understanding Floppy Disks" elsewhere in this issue.) A CP/M-based system may have 5-inch floppy-disk drives with about 90 K bytes of memory capacity per drive or a hard disk with up to 30 million bytes capacity. I won't advocate that everyone get the maximum amount of mass storage.

If you are the only one using your word processor, I can't imagine the need for a hard disk. A library of floppies would probaby be more practical. Whether you need the increased capacity available on 8-inch floppy disks depends on your use of the system. Will you have many documents or relatively few? How long will they be? Don't forget that you should have backup copies of each, as well as the current versions. If you're not used to thinking of the number of characters in a document, it might help to know that this article has approximately 44,360 of them (not including the tables and figures). If you plan many small documents, such as letters, you may run out of directory space before you run out of

storage capacity, since in some systems a CP/M disk may have only sixty-four files on it.

Regardless of your decision on 5- versus 8-inch drives, I have two suggestions: unless you have a good reason not to, get a two- (or more) drive system. This will double your on-line capacity, and more importantly, it will enable you to quickly copy files for backup purposes. The other suggestion is to get double- or quadruple-density drives if you can afford them. This will increase your storage capacity and speed up data reading and writing at a small marginal cost.

A word-processing program provides the editing and printing functions in one package.

Printer

Your printer choice will be determined by the things you plan to do with your system. Personal computer printers come in two types: dot matrix and formed character. In the case of a dot-matrix printer, each character is made up of a pattern of dots. The most common grid size is 7 by 9 dots. A formed-character printer works like a typewriter, striking type against a ribbon.

Formed-character printers are generally slower than dot-matrix printers. They typically print at a rate of 55 characters per second (cps), while dot-matrix printers print at 150 cps and up. Formed-character printers are also more expensive than dot-matrix printers. What do you get for the extra investment? Print quality.

Print quality in a formed-character printer is about the same as that of a good office typewriter. With a carbon ribbon and proportional spacing, the results look professional. You won't convince any of the 3000 people on your mailing list that they've received a per-

Word Processor	Editor	Print Formatter
Auto Scribe	20	19
Electric Pencil	30	24
Magic Wand	49	76
WordStar	69	48

Table 1: *Number of features of the four word processors under review. The author looked for the presence of 164 features (too numerous to list in this article). This gives a* rough *measure of each system's power, but the results can be misleading. Many of the functions done with a single command on one system may be done using two or more commands on the others, and complexity can intimidate a beginner and slow down even an experienced user.*

sonal letter if you use a dot-matrix printer.

The formed-character printers on the market today allow for a good deal of control. The computer can usually move the print head 1/120 of an inch horizontally and 1/48 of an inch vertically. This means that *boldface* printing (ie: overprinting with a slight shift in position), superscripting, and subscripting are possible. Character pitch and line spacing may be set to very fine tolerances and proportional spacing of characters is possible. A formed-character printer allows the program to justify text by making slight adjustments in the spaces between letters, rather than inserting blanks between words. They can even back up, allowing for printing multiple columns on a page.

Are you going to use your system for fancy business letters or for writing important proposals? Will you be using it as a composer, creating camera-ready copy to take to the printer? If so, you will need a formed-character printer.

Terminal or DMA Video

A final hardware choice has to do with the manner in which the keyboard and video display are built for your system. A terminal is an all-in-one device. It has both a keyboard and a video display. [*A video display is sometimes referred to—incorrectly— as a* CRT, *which stands for "cathode ray tube." A video display contains much more than just the screen itself....***ed**] The computer displays information on the video display by writing it from its memory to the terminal memory. This transfer goes at a high rate (at least 960 cps), but in computer terms this is still slow. In the case of direct memory access (DMA) video, the computer changes the screen display by merely changing the contents of its memory or changing the values in a few internal registers. This is much faster and the difference is noticeable, though not overwhelming, when doing word processing.

Not surprisingly, terminals do offer some advantages over DMA systems. They've been around for many years and are sold in vast quantities in the industrial world. Thus they are standardized and have established maintenance sources. There are well-established, highly competitive channels of industrial distribution, so discounts are readily available. You can also shop around until you find one with a keyboard you like.

Regardless of which method your system uses for displaying information, you'll need a keyboard for entering it. If you have a DMA system, some provision will have been made for an independent keyboard. Some DMA-based systems come with built-in keyboards—for example, the Radio Shack computers. In other cases, you will need to get a detached keyboard, or order the optional keyboard that is offered on a number of printers. In either case, your range of choice will be limited.

Don't take this choice of keyboards lightly. Make sure you like the feel of the keys. If you are an experienced typist, be sure the keys are where you like them to be. You'll be using the "control" key very often in word processing. Be sure that it is well located. Finally, a small but crucial feature: make sure that your keyboard repeats a character after the key has been held down for a half second or so. This is most useful during editing.

Software Again

Software is the most important step in putting together a word-processing system, so be sure that whatever hardware you settle on is compatible with the program you select. If you buy from a dealer, make certain that he will assume responsibility for getting the software running with your system. If you do it on your own, the software manufacturer should be contacted to be sure that the hardware you plan to buy is compatible with their program. The ideal situation is to get hands-on experience with your hardware and software before buying it, but that isn't always possible.

Program Comparisons

With this background information in mind, let's take a look at the four programs. I have used each of them for at least a month and have used them all for real work. A detailed comparison on 164 features was made, and the following summarizes that study. (See table 1.)

Auto Scribe

Auto Scribe has been designed for the computer novice, and has been kept relatively simple. The manual assumes no knowledge of computers. It is the only package

in which the *operating system* (ie: the internal "controller" program in a computer that takes care of the many technical details involved in computer operation) is even hidden from the user. When the system comes up, the operator is presented with a "menu" list on the screen, and may choose to create, revise, or view a document or perform disk operations. Disk operations include copying files, deleting files, displaying directories, etc.

Creating, revising, and viewing of documents differ only in the way files are handled. In creating a document, a brand-new file is produced. In revising a document, an old file is revised and a second file with a new operator-assigned name is created. When viewing a document, it may be altered, but no disk file changes are made. In all cases, the operation of the editor seems to be identical.

When working with a file too large to be contained in the memory buffer, portions of it must be kept on the disk and brought into memory when needed. In keeping with the goal of serving the computer novice, these disk-to-memory transfers are handled automatically in Auto Scribe. Once a portion of a document has been automatically written on the disk, it is impossible to go back to it without restarting the operation.

The editor and print formatter are integrated into one program. A document is printed by turning the printer on while scrolling through it a screen at a time. The editing philosophy leans toward on-screen formatting prior to printing, though a number of formatter commands are used.

Electric Pencil

Electric Pencil is the "granddaddy" of personal computer word processors, having been available for several years. Like Auto Scribe, the system is designed for easy use by the novice. When the system is activated, it is in editing mode and

Word Processor	Editor Buffer Size (in bytes)
Auto Scribe	32 K
Electric Pencil	46 K
Magic Wand	50 K
WordStar	31 K

Table 2: *Editor buffer size. This is the amount of memory space set aside by the program for text. Magic Wand has the largest memory area for text during editing. This is achieved by using a separate program for the print formatter—thus the print formatter does not take up valuable memory space during editing. Because of this Magic Wand avoids the long delays that would be necessary for disk transfers if the document being edited is too big for memory. On the other hand, you are not able to see the final page and line lengths while editing because the formatter is not active. To do this you must leave the editing program and run the formatter.*

it is possible to enter and revise text. In order to print material or perform disk operations, you strike a control key to switch modes. To return to editing after printing or disk operations, you merely strike the escape key.

The print formatter and the editor are fully integrated. At any time it is possible to stop editing and print all or part of the text or to stop printing and resume editing.

When a document is revised, you must explicitly save the new version. You are free to rename it, but if you use the same name, no backup copy is kept. If a document is too large to reside in memory at one time, it must be broken up into several smaller documents since Electric Pencil makes no provision for such cases.

Electric Pencil is able to operate only on systems using DMA video displays. Each of the others is capable of using a terminal as well.

Magic Wand

Magic Wand, like WordStar, is considerably more ambitious than either Auto Scribe or Electric Pencil. It has more editing capability than either of the former packages, and many more print formatter features. The editor and main print formatter are two completely separate programs; however, the designers of Magic Wand have compromised by including a rough-draft print routine in the editor.

You bring the system up by running either the editor or print formatter. The editor comes up in an edit mode which is similar in facility and style to that of Electric Pencil; however, by switching to an extended edit mode, you have a much wider range of capabilities, particularly in the area of examining text which has been prestored on the disk and selectively including it in the document. You may also make draft copies of all or any portion of your document while in the extended editing mode.

When a file is revised, the operator has the option of giving it a new name or using the old one. If the old name is used, the new version is saved automatically and a backup copy of the original remains on the disk. The second time a file is revised, the first original is automatically deleted.

Documents which are larger than available memory are allowed, but you must explicitly save all or part of memory when it becomes nearly full. Once saved, you cannot back up to it without rerunning the editor. However, Magic Wand has the largest memory buffer of any of these systems. (See table 2.)

When running the print formatter, the operator may automatically print a file, set or change any print control variables (such as margins and spacing), or input data to determine what will be printed. In fact, the operator is free to issue any of the formatter commands available on the system.

WordStar

WordStar is also a very ambitious program. It has more editing features than Magic Wand, but fewer print formatter features. The editor and the print formatter are run as one program, as with Auto Scribe or Electric Pencil; however, they are not as tightly integrated, because a file must be saved on the disk before it can be printed. While it is not possible to print the file being edited without first taking the time to save it on the disk, keeping the editor and formatter as one large program is a good idea, since it is possible to preview margins, page breaks, and line lengths during editing. Word Star, like Auto Scribe, leans toward formatting text as it will be printed, and away from using embedded commands.

When WordStar comes up, it is in basic editing mode, like Magic Wand. In order to perform other editing operations, disk operations, or print, it is necessary to enter one of five subsidiary modes. This is a bit overwhelming, especially if you are not experienced with computers. To help deal with this complexity, Word Star displays "help" menus at all times. These may be suppressed once you become proficient, and subsidiary menus do not appear at all unless you pause for a second or two while deciding what to do.

As in Magic Wand, it is possible to either give a file a new name after it has been revised, or to save it under the old name with a backup copy. It is possible to edit documents larger than memory, and the system handles transfers to and from the disk automatically, as in Auto Scribe. This is a good feature and there is no limitation on being able to back up when the early part of the file has been saved.

Comparison Summary

As mentioned above, I compared

these four programs based on 164 features. This section summarizes those comparisons in the general areas of documentation, ease of use in learning and editing, and printing power.

Documentation

The first thing any word-processor user encounters is the system manual. At first it functions as a teaching tool. Then, after the operator has learned to use the system, it is a reference. The prompts and diagnostic messages that the program provides while it is running are also a part of the documentation. In looking at documentation, we must remember that Auto Scribe and Electric Pencil are relatively simple compared with Magic Wand and WordStar, so the authors of their manuals had easier jobs.

The Electric Pencil manual is clear and well-organized. It will not intimidate a beginner, and serves as a perfectly adequate teaching tool for the new user. It is poor as a reference because it has no index; the reference card summarizes the editing commands but not the print commands or disk operations.

The Auto Scribe manual is designed for the beginner, and since the system is relatively small, it could have been as good as Electric Pencil's manual, but it isn't. It is poorly organized and verbose. Essentially identical descriptions of the editing commands are given three times, under the creating, modifying, and viewing of documents. More is certainly less in the case of a teaching manual. As a reference, it is better than Electric Pencil because it has an index.

The Magic Wand manual is unique. It is typeset, bound, and contains many screen photographs. That alone makes it impressive and inviting to the beginner. The author of this manual had a rough job. He had to teach the beginner about complex print for-

Using the Apple Pascal Text Editor As a Word Processor

by Jef Raskin

While not designed as a general-purpose word processor, Apple's version of the UCSD Pascal system's text editor does a creditable job. Apple Computer Company uses the system to produce all of their manuals. Some outstanding examples are the new, spiral-bound *Apple II Reference Manual*, in which the output device was a phototypesetter, and the *Apple Pascal Reference Manual*, in which the output device was a Qume printer. Dr Bowles' books on Pascal were written on this editor, and it is used by Personal Software (who produce MicroChess and VisiCalc) as well.

As a text editor *per se*, it is fair-to-good in terms of features. To name a few of its abilities, it will:

- align margins left or right, or center lines
- automatically put in returns as you type
- indent (or outdent) paragraphs
- allow you to insert or delete freely
- move paragraphs (or any portion of the text) from place to place
- allow you to undo a deletion
- remargin and reparagraph if necessary
- do a wide range of search and search-and-replace operations

Where it excels is in its command of the Apple/UCSD filing system. You can pull in parts from many disk files to make a new document, or send parts of an existing document to different places. Programs and text are edited by the same editor, so you need learn only one set of commands to do program editing as well as word processing. This is especially convenient when you want to embed extensive documentation in a program or have a program appear in some other document.

The main drawback of this system is that it does not have a very good human interface. While it is average in this regard when compared to some of the other personal computer word processors, it could be much better. Even though Apple provides a fine reference manual, which has more than enough information to allow you to use this editor, beginners may find it difficult to get started. A tutorial manual would be much appreciated.

One last feature that I especially enjoy: when you turn the machine on in the morning, and you wish to continue working where you left off the night before, the computer remembers your place—all you need to do is type the letter "E" (for Edit) and there's your document just as you left it.■

matting and programming, as well as the more straightforward editing and printing features, and he has done an excellent job. Two thirds of the manual is devoted to leading the user through a series of lessons. These utilize increasingly complex sample text files which are included on the disk. This is a unique and successful approach. The last third of the manual is a reference guide, and it is clearly written; however, the index is incomplete, so I have wasted much time skimming. The Magic Wand reference card is excellent and nearly complete. If it included an index into the manual, Magic Wand would have been the best reference manual.

The WordStar manual is a thick loose-leaf book. It is well-written and quite complete, but it's a bit overwhelming even to an experienced computer person. It would intimidate a novice. Ironically, it is full of typographical errors, and like all but Magic Wand's documentation, it is a printout composed using the system.

The authors of WordStar also faced a difficult task in teaching the beginner to use its complex editing facilities. Rather than writing a teaching-oriented manual, they have provided extensive menu prompting during editing. The top half of the video display is used to display brief explanations of the meanings of available commands. When you change modes, you get a new prompt screen if you take more than a second or so to give a command. This is a unique and excellent approach. At first, you're overwhelmed by the explanations, but you soon learn to use them. Later, when you no longer require them, they disappear. The experienced user eventually suppresses them altogether.

As a reference, the WordStar manual could also have been improved substantially by the inclusion of an index. There is a list of control commands with their page numbers, but no index and no reference card.

Auto Scribe rates ahead of the others as a reference manual, simply because the index is more useful. The others could easily improve. On the other hand, as a teaching manual, only Auto Scribe is notably insufficient. Both Magic Wand with its lessons and practice files and WordStar with its on-line help facilities are innovative and effective.

Ease of Learning

As with documentation, the simplicity of a system makes it easy to learn. Electric Pencil is the easiest to learn. Cursor movement, text entry, and revision are straightforward. The conceptual separation of the print- and disk-handling subsystems is clear and

Electric Pencil is the "granddaddy" of personal computer word processors.

sufficient. My eleven-year-old daughter was able to use Electric Pencil to create, revise, and print simple documents after about fifteen minutes of instruction and practice.

Auto Scribe is more difficult to learn to use because the text is presented in screens during editing, rather than as one continuous document, and because separate modes are required for cursor movement, inserting characters, and deleting characters. In addition to being somewhat confusing, these restrictions inhibit learning because they slow down editing.

The basic functions of the Magic Wand editor are as easy to learn as those of Electric Pencil. I gave a secretary a twenty-five-page manuscript to type, along with the Magic Wand reference card and a five-minute demonstration. She was able to input the manuscript, revise it, and print a draft copy without asking a single question. The advanced editing features which deal with blocks of text, searching and replacing, and examining and including information in disk files are also clearly designed and easy to learn.

The advanced print formatting and programming features are more complicated and therefore more difficult to learn. The lessons mentioned above are very good, but they take time. An experienced programmer would probably read the lessons without actually running them, but would still require time to become proficient in the use of these features.

These advanced features may also be used to build custom applications, which can be used by complete novices. Since Magic Wand provides much facility for operator interaction, it is possible for an advanced user to set up the system so that it prompts the operator, telling him exactly what he needs to do. There is no analog to this capability in any of the other systems.

WordStar is the most difficult of these systems to learn to use. As mentioned above, the designers of WordStar have leaned toward formatting material as it will be printed during editing, rather than embedding print commands in the text. As such, they have chosen to provide many editing commands, and have been forced to include others. Therefore, for even simple editing, there is more to be learned than with any of the others. Many of the simple editing commands require two keystrokes rather than one. The complexity is mitigated by the help menus, but it is still there.

Because of reliance on formatting during editing and the omission of programming features, the print formatter commands in WordStar are easier to learn than Magic Wand. They are harder to learn than Auto Scribe or Electric Pencil because there are more of them.

For editing, Electric Pencil and Magic Wand are easiest to learn,

Auto Scribe is next, and WordStar is most difficult, because there is so much to learn. Learning to use all of the advanced programming and formatting features of Magic Wand will be more difficult than learning Auto Scribe. Finally, a novice could easily learn to use Magic Wand for sophisticated applications if an advanced user had programmed it with prompting ability.

A Note About Auto Scribe
We were informed at press time that Micro Source will soon be marketing a new, greatly enhanced version of Auto Scribe for CP/M systems. (The Auto Scribe package reviewed in this article is actually a North Star Horizon version.) The new package will feature boldface, superscripting, subscripting, block move, block copy, double underlining, etc. Not all details about the system have been unveiled, but the enhanced version is expected to be competitive with the Word Star system.

Editing Power

All of these editors allow you to key in text, to make insertions and deletions, to scroll the screen over the text in memory, and to move the cursor around on the screen. They also allow you to search through the text for a specified character pattern and possibly replace it with another. In some of them, you may mark blocks of text in order to move, copy, or delete them, or you may include blocks of text which have been stored on the disk in your document. Each system also allows you to control the display on the screen to some extent and to control the naming and backing up of files after they are created or revised.

Table 3 summarizes editing power of the four systems. In general, the ratings are based upon the number of features available in each category. Where the function of a command in one system cannot be achieved even by using combinations of commands in another, or if it can only be done very awkwardly, it was subjectively weighted higher.

A quick glance at the table shows that WordStar has more features than any of the others. Auto Scribe is ranked above it in insertion/deletion power because of the unique way in which deletes are performed. It is possible to delete from the cursor position through the next occurrence of a designated character. In addition, deletions are not executed immediately, but are displayed in lowered intensity until the operator verifies them. Magic Wand enjoys a substantial advantage over the others in its ability to selectively examine and include material stored in disk files. The user is able to create his own help menus, which guide the operator in the selection of text for insertion into the copy being edited.

Editing Speed and Ease

While WordStar has more editing features than any of the others, it is not necessarily the fastest and easiest to use since many of the editing commands require two keystrokes and can be simulated using multiple commands in the other systems. If a two-keystroke command is used and the operator hesitates for a second or so between keystrokes, the system gives a help screen, which takes time because the disk is read.

WordStar also slows down considerably when its memory buffer becomes full. As table 2 shows, WordStar has a relatively small memory buffer. This is part of the price paid for having the print formatter and editor in memory together. As mentioned above, WordStar is fully automatic in transferring text to and from the disk if the document is too large to be held in memory at one time. This simplifies your conception of the system and enables you to back up to the start of a file even if it has been written onto the disk; however, editing is slowed for files which are over 31 K bytes of memory in length.

There are also questions of style, which are highly subjective. Auto Scribe and WordStar both restrict cursor movement to portions of the screen where there is text, while Electric Pencil and Magic Wand allow it to go anywhere. With Electric Pencil and Magic Wand, you have conceptual clarity—the cursor never surprises you by where it ends up—but you pay the price of needing extra keystrokes at times to get it where you want it. These extra keystrokes might be exasperating to people whose keyboards don't repeat when keys are held down. Electric Pencil is roughly equivalent to Magic Wand, but Magic Wand can perform many editing functions that Electric Pencil cannot. Therefore, Magic Wand ranks above it. Auto Scribe has to rank at the bottom of this category since it has the fewest features and they are awkward to use. Insertion, deletion, and cursor movement are all carried out in separate modes. It is also slow for large documents, when it is necessary to read and write the disk. Unlike WordStar, it is not possible to back up to material once it has been saved. Like WordStar, Auto Scribe attempts to preview margins, line, and page sizes, but doesn't do it as well.

Because of stylistic differences and differences in document sizes, it is impossible to come up with a

Feature	Auto Scribe	Electric Pencil	Magic Wand	Word Star
Cursor Movement and Scrolling	3	2	2	1
Block Operations	N	2	1	1
Auxiliary Disk File Operations	3	4	1	2
Search and Replace	2	3	2	1
Insert and Delete	1	3	3	2
Screen and File Control	3	3	2	1

Table 3: *Editing power rankings, shown by categories. "1" indicates the highest ranking; "N" means "none available." In some cases two programs tied for third.*

strict ranking of the programs. At times the power of WordStar will make it the fastest and easiest to use, but in other cases Electric Pencil or Magic Wand will beat it.

Table 4 summarizes the printing power of the various word processors. Again, the ratings are primarily based on the number of features in each category.

The page layout features have to do with controlling the physical size of the page and margins. Page and paragraph control have to do with headings, footings, page numbering, and automatic indenting and spacing between paragraphs. Line control refers to various sorts of justification and options for spacing between lines. Type control includes special font shifts, like boldface and double striking, as well as spatial shifts like subscripting and proportional spacing.

Programming features, which are found only in Magic Wand, are the inclusion of variables, simple assignment statements, and conditional branching. Variables may get values either from the operator, from a disk file, or by assignment. Magic Wand and WordStar are also able to write their formatted print output to disk. This capability complements the programming features of Magic Wand and it is more powerful than WordStar.

The printer control features have to do with operator control over the printing process. Such things as starting and stopping printing and issuing formatter commands at print time are included in this category. I also considered eight miscellaneous features.

As WordStar dominates in the number of editing features, Magic Wand dominates in the area of print formatting. This is partially a reflection of the differing philosophies regarding formatting during editing versus using embedded commands. There are many things which Magic Wand and WordStar are able to do which cannot be done with Auto Scribe or Electric Pencil.

Who Are You?, Revisited

With these comparisons behind us, let us return to the four hypothetical users described earlier.

The author usually creates and revises relatively large documents, so less emphasis is placed on printing than editing. Any word processor obtained will be far superior to a typewriter, and I don't think that an author can go wrong with either Electric Pencil,

Feature	Auto Scribe	Electric Pencil	Magic Wand	Word Star
Page Layout	3	3	1	2
Page and Paragraph Control	3	3	1	2
Line Control and Justification	3	3	1	2
Type Control	3	3	1	2
Programming	N	N	N	1
Printer Control	3	2	1	3
Disk Output	N	N	1	2
Miscellaneous	N	3	1	2

Table 4: *Formatter power rankings, shown by categories. "1" indicates the highest ranking; "N" indicates "none available." In some cases two programs tied for third.*

Magic Wand, or WordStar. If it is possible to adjust to its style and complexity, WordStar will provide the most powerful editor. Also appreciated will be the fact that words and lines are in the same position on the screen and printout when revising from marginal notes. This makes it easier to find one's place.

If documents are consistently long, it will be necessary to break them into parts or an author must learn to live with the slowdowns due to the limited memory buffer size with WordStar. If the author doesn't like the WordStar style, I would advise the use of Magic Wand. The editor is more powerful than that of Electric Pencil and it is faster than WordStar for documents which are between 31 K and 50 K bytes.

The marketing manager is concerned with printing multiple copies of similar documents—for instance, in doing mass mailings. The manager does relatively little editing but needs print-time power. This is the clearest choice of all. The programming features of the Magic Wand print formatter and its flexibility at selecting information from the disk during editing are designed with this person in mind. Futhermore, the capacity of Magic Wand for presenting user-defined help screens and operator prompts makes it possible for an experienced user to set up specialized application packages that can be used by clerical workers with no training. Even if the marketing manager prefers the editing style of WordStar, I would advise the use of Magic Wand.

The secretary comes next. In doing general correspondence, many short unique documents are created. Much editing is done, but not as much revision as the author. Complex print formatting features aren't really necessary for this sort of work. Any of these systems can be used. The relative simplicity of Auto Scribe and Electric Pencil will be appealing; however, the editing power of Magic Wand or WordStar will probably make a secretary more productive once their use is learned. On the other hand, being able to print while editing, without saving the text on the disk (as in Auto Scribe and Electric Pencil), will make the job go faster.

The newsletter composer works with relatively large documents, but doesn't revise them as frequently as an author. Thus, the editor will be less important here than it is to an author; however, the print formatter will be much more important.

This choice should be limited to either Magic Wand or WordStar because of their print formatting capability. The choice will hinge on some of the same factors as the author's. Editing style, power, and disk speed will be important. The composer will tend to lean toward WordStar for the ability to see page and line breaks on the screen during editing. To do the same thing in Magic Wand, the text must either be printed or written on the disk in its final form, either of which requires running the separate print formatter program. On the other hand, the formatting power of Magic Wand will be appreciated, and also its capacity for interaction during printing, for example, to make quick changes in pitch and line spacing in order to stretch or compress a column. This is a difficult choice, but happily, both alternatives are good.

In short, all of these systems have advantages and disadvantages. Armed with the information in this article, I hope you will pick the word processor that's right for you.■

About the Author

Larry Press is president of the Small Systems Group, a software evaluation organization in California. He has been involved with personal computing since its inception. With a PhD in information processing, Larry has taught computer science and business administration at various colleges around the country. His most recent project has been a detailed evaluation of word processors, out of which this article grew. Readers wishing to obtain more information about the full report should contact the author at the following address: Larry Press, c/o Small Systems Group, POB 5429, Santa Monica CA 90405.

ALMOST PERFEC.

The MAGIC WAND™ is the most powerful, most flexible, most reliable, *most usable* word processing software available for a CP/M®-based computer.

That's not bragging. That's just telling it like it is.

The MAGIC WAND *is* the best word processing software ever written for a microcomputer. It can do more work in less time with higher quality than any other product you can buy.

The MAGIC WAND is a rock solid piece of software. The command structure is simple and logical and complete. We have not tossed in features without thought to the overall design of the package. Nor have we included any feature that is not thoroughly implemented. The programs are crash-proof and completely reliable.

And the system is supported by what we are told is the best user's manual ever produced for microcomputer software. It contains a step-by-step instructional program designed for the novice. The trainee uses sample files from the system disk and compares his work to simulated screens and printouts in the manual.

Support doesn't stop when you buy the package. As a registered user, you receive our bi-monthly newsletter which answers questions, reports upgrades and teaches new applications of the MAGIC WAND.

It's through a lot of hard work that we are able to offer you a product that is "almost perfect," but we aren't about to stop working until we can say that the MAGIC WAND *is* perfect.

Full screen text editing

The MAGIC WAND has probably the most responsive and easy-to-use editor available for either a serial or DMA terminal. It uses only single stroke control keys to give command and takes advantage of the special function keys on your terminal whenever possible. In addition, you can set up library files with coded sections that you can merge by section name.

Full text formatting commands

The MAGIC WAND allows you to set the left, right, top and bottom margins, page length, indentation, paragraph indentation, (incuding "hanging" paragraphs), text left flush, right flush, justified (two ways), literal or centered, variable line and pitch settings, variable spacing (including half lines), bold face, underlining (solid or broken), conditional hyphenation, sub- and superscripting. You may change any of these commands at run-time *without reformatting the file.*

Merging with external data files

You may access any external data file, with either fixed length or sequential records. The MAGIC WAND converts the record into variables that you define and can use like any other variable. Of course, you may use the data for automatic form letter generation. But you can also use it for report generation.

Variables

You may define up to 128 variables with names of up to seven characters. The current value of a variable may be up to 55 characters, and you may print it at any point in the text without affecting the current format. Although the MAGIC WAND stores the variables as strings, you may also treat them as integer numbers or format them with commas and a decimal point. You may increment or decrement numeric variables or use them in formatting commands.

Conditional commands

You may give any print command based on a run-time test of a pre-defined condition. The conditional test uses a straightforward IF statement, which allows you to test any logical condition of a variable. You may skip over unneeded portions of the file, select specific records to print, store more than one document in a single file, etc.

True proportional printing

The MAGIC WAND supports proportional print elements on NEC, Diablo and Qume printers. Other formatting commands, including justified columns, boldface, underline, etc., are fully functional while using proportional logic.

Available on 8" soft-sectored and 5 1/4" Northstar or Micropolis (hard or soft sectored) diskettes, as well as ONYX hard disk. Terminals supported include—ADDS, Beehive, Cromemco, Dynabyte, Hazeltine, Heath, Imsai, Intertec, Lear Siegler, Microterm Act V, Perkin Elmer, Sol VDM1, Soroc, TEC, TEI, Televideo, TRS80 Mod II, Vector Graphics, plus a variety of video boards.

Also available in OASIS

small business applications, inc.

3220 Louisiana • Suite 205 • Houston, Texas 77006 • 713-528-5158

CP/M is a registered trademark of Digital Research Corp

Circle 140 on inquiry card.

Text continued from page 27:
system might be 8 K bytes long. To load in such a program from cassette can take anywhere from 4 to 15 minutes, and if any one of the bytes is missed or read in wrong, then the whole 8 K bytes must be read in again.

On the other hand, an 8 K-byte program stored on a program cartridge may be loaded into an Atari computer in a few seconds. Plugging in a program cartridge is easier than placing a cassette into a cassette player or a disk into a disk drive. When you want to insert a program cartridge, you lift the lid to the program cartridge compartment. Lifting this lid automatically shuts off the computer, so no current is live anywhere near your fingers when you reach in to replace one cartridge with another.

Nor do you have to treat cartridges with extreme care, especially when compared to the care that must be exercised with magnetic media. An entire disk or cassette can be made unusable by one fingerprint or a stray magnetic field, not to mention the owner who unintentionally erases a cassette or disk by recording over it. But program cartridges store data in the form of read-only-memory circuits, which are unaffected by magnetic fields. The cartridges themselves are almost totally sealed, protected from dust and fingerprints. The program cartridges allow the computer to be a mathematics tutor at one moment, a basketball game the next, and a computer that runs programs in BASIC after that. Such cartridges may be the most practical medium for program storage in a consumer computer.

Software

I have already mentioned the high quality of video game cartridges for the Atari 400 and 800. Many more such cartridges are becoming available, and I assume that they will be very good. But there's more to computer software than games. What happens when you remove the game cartridge from an Atari computer and turn it on? Is there any software left? Does the computer shut off? What happens when you turn on the computer without first placing a program cartridge inside?

Fortunately, the company has done some careful thinking about these questions. When the computer determines that no program cartridge is in place, it activates a program that is always available, because it's part of the Atari's permanent memory. It's called NOTEPAD, and it lets you place letters, numbers, and special graphics characters wherever you like on the television screen. You can type messages on the screen, intermixed with elaborate patterns and pictures.

NOTEPAD may not strike you as particularly impressive (although your children may be thrilled with the idea of putting messages onto the television screen), but this little performance demonstrates a lot of care and development on the part of Atari. If nothing else, NOTEPAD tells you that the Atari system is working and displaying properly on the television screen. More significantly, since there is no program cartridge in place, NOTEPAD shows you some of the capabilities of Atari's *operating system*: the software that is part of Atari's permanent memory.

Features in the operating system are an integral part of the Atari computer, as much as the colors in the display, or any other hardware feature. The operating system is very important, because it provides the means for the user to communicate with the computer. The operating system *reads* the keyboard and the joysticks, and it *writes* to the television screen. Because the operating system makes it easy for the user to communicate with the computer, the computer becomes a useful tool.

Text Entry

The text-entry system on the Atari computers is well designed and well executed. You type, and your characters appear on the screen. Initially, the keyboard yields only uppercase letters, but pressing the CAPS LOCK key unlocks the shift. The keyboard then behaves exactly the way a typewriter keyboard behaves: shifted keys yield punctuation marks and uppercase letters; unshifted keys yield numerals and lowercase letters.

Other personal computers have keyboards that look, but fail to act, like typewriters. On some systems, the shifted keys yield lowercase and the *un*shifted keys yield uppercase. On some other systems, the left and right shift keys yield different characters even when used with the same letter keys!

The text-entry system on the Atari computers lets you enter uppercase and lowercase text as easily as if you were typing. If you make a mistake, you can back up and delete with a single keystroke.

Some of the keys have arrows on them. These arrows enable you to move a cursor about the screen. The cursor is in reverse intensity from the character underneath it, so there's no mistaking the cursor's current location—it stands out as a white box on a black background, or vice versa. The

Atari lets you learn quickly . . . to make mistakes . . . and to recover from them gracefully.

cursor indicates the place on the screen where the next character will appear.

It's extremely easy to move the cursor about the screen, because the consequences of any keystroke are immediately visible on the screen, and any such consequences can be undone with another keystroke. If you make a mistake, you can correct it immediately or some time later. Nor are you limited to editing a single line: you can move the cursor about and edit anything on the screen—a definite plus.

Not only can you put the cursor anywhere on the screen and overstrike anything, but you can also make room for new characters by pressing CONTROL INSERT. You can also close up space to delete unwanted characters by pressing CONTROL DELETE. These character insert and delete functions are very powerful and are made available to the user in a clean keyboard configuration. By sitting at the keyboard and typing for a few minutes, you can learn how to talk to the Atari. Atari lets you learn quickly and painlessly, in a manner that allows you to make mistakes and to recover from them gracefully.

Because these screen/keyboard editing functions are part of the operating system, they are always available to you, whether you want to write a program or simply enter data into a program that has requested it. Now that you know how to communicate with the Atari, you will probably do so in BASIC.

BASIC

BASIC comes with the Atari 400 and 800, in the form of a plug-in program cartridge. In most ways it is like most BASICs available on personal computers, performing mathematical and trigonometric functions with nine-digit precision, and understanding such statements as LET, PRINT, RUN, CSAVE, CLOAD, IF . . . THEN, FOR . . . NEXT, PEEK, POKE, and others. The aspects of Atari BASIC that differ from most other BASICs are the special commands to control color graphics and the audio channel to your television set. Simple commands such as "PLOT (X, Y)" and "COLOR = " let the BASIC programmer put any character at any X,Y coordinates on the screen, and in any color. [*In this regard, Atari BASIC is similar to BASIC on the Apple II computer....***ed**] Similar commands let the BASIC programmer generate sounds for the television speaker, without the need to PEEK and POKE special memory locations with special values. (PEEK and POKE are BASIC commands that let you examine and change the contents of memory directly from a BASIC program. Some computers require this somewhat unwieldy method for graphics programming.)

It's one thing to have a fancy color display, but quite another to

Atari BASIC Isn't Perfect (Even Though It's Close . . .)

Most of the major microcomputers—the Apple II, the Radio Shack TRS-80, the Commodore PET, the Ohio Scientific Challenger computers—use basically the same dialect of BASIC, written by Microsoft. Although each has some extensions particular to that individual computer, programs can often be written so that they will execute without modification on any of the above computers.

Unfortunately, Atari has provided its own BASIC version that, for all its wonderful graphics, color, and game commands, will make the conversion to Atari BASIC moderately difficult. In particular, there are three differences between Atari BASIC and most other BASIC dialects—two are minor and one is major.

The first minor difference is that subscripted variables cannot be used in READ or INPUT statements. For example, in Atari BASIC, to input a number from the keyboard into array variable A(3) (an array is an orderly arrangement of numbers, characters, and/or symbols), you have to write:

```
120 INPUT X
130 A(3) = X
```

Writing INPUT A(3) in an Atari BASIC program is not allowed. It will generate a syntax error. (The numbers 120 and 130 are program line numbers.)

The second minor difference is that Atari BASIC allows arrays up to only two dimensions. Thus, A(3,2) is valid in Atari BASIC, but A(3,2,2) is not. (Microsoft allows arrays of any number of dimensions, although it is only occasionally that a program uses an array of three or more dimensions. Still, that third dimension will not be available the one time you *do* need it.)

The third, and major, difference is that Atari BASIC does not allow string variables to be dimensioned; in addition, its notation for substrings will confuse people familiar with Microsoft and other BASICs. For example, in Microsoft BASIC, you can dimension a string variable A$ to have five entries and each can be up to 256 characters long. This can be done with the statement:

```
10 DIM A$(5)
```

You can then make assignments such as A$(1) = "THE ANSWER IS" and A$(2) = "THERE ARE NO ANSWERS TO THIS PROBLEM".

Atari BASIC allows you only *one* variable with name A$. It must still be dimensioned, but the above DIM statement now means that A$ is a string variable that can have a maximum of five characters. So, if A$="ATARI", then A$(1) refers to the first character of the string, "A", A$(2) refers to the second character, "T", and so on. Also, A$(3,5) refers to the substring of A$ given by the third to the fifth characters. This is quite different from Microsoft BASIC, where A$(3,5) refers to the string that is in the third row and fifth column of a two-dimensional array variable A$.

Other aspects of this fundamental difference in string handling between Atari and other versions of BASIC will not be covered here. But, with these differences, Atari has perpetuated the confusion of novice (and not-so-novice) programmers by presenting them with yet another version of BASIC that doesn't act the way they think it should. . . .**ed**

give even the beginning programmer easy access to that display and control of its appearance. Atari has managed this feat. Without software to put the reins of power in your hands, even the most powerful hardware isn't of much use.

Incidentally, the Atari manual shows you how to write *chained* programs. When a program is too big to fit in your computer's programmable memory, you can break it into two or more segments; each one is small enough to fit in the available memory. The Atari computer will then load the next program segment from cassette or disk and run it. Chaining programs lets you surpass some of the limitations of your computer's memory. (And no matter how much memory your computer has, sooner or later you'll write a program too big to fit inside it!)

Peripherals

The *peripherals* (ie: the printer, cassette recorder, and disk drive) connect very easily, more like the way stereo components interconnect than the way most personal computer systems do. Instead of soldering wires or pushing a large circuit card into a socket, you need only plug a cable from the peripheral device into a socket on the Atari marked *Peripheral*. If you have another peripheral device, you plug it into the first one. The plugs and sockets are all unique so, for instance, you can't plug a power supply cord into a video socket. They are also labeled and asymmetric, so you can't put a plug into a socket upside-down. This is perfectly reasonable and what you would expect from a good consumer product; but it's quite unusual for personal computers.

JADE. INFLATION FIGHTER FOR THE 80's.

KNOCKOUT PRICES

ROUND #1.

JADE has been selling low-cost name brand merchandise for more than 5 years. In that time we built a long list of satisfied customers for our mail order computer products as well as making JADE a name respected throughout the nation. Order your computer systems directly from JADE's full line including Apple products. Jade carries an extensive selection of other supplies for the electronic/computer lab.

Leedex

VIDEO-100

12 Inch Monitor

Have you noticed your eyes beginning to water after several hours in front of that converted 1939 Atwater-Kent Kinescope you're using for a monitor? Well, maybe it's time to get a real video monitor with 12 MHz bandwidth, reliable solid-state circuitry, and a 12 inch (diagonal measure) screen. Front panel controls include power, contrast, brightness, horizontal hold, and vertical.

VDM-801210 ..... **$139.00**

Special Package Price

ROCKWELL AIM-65

The Head-Start in Microcomputers

KIM-1 compatible
On-board printer
Full ACSII keyboard
AIM-65 w/1K RAM..$395.00
AIM-65 w/4K RAM..$439.95
8K BASIC ROM..$100.00
POWER SUPPLY..$59.95
CASE for AIM-65..$49.95
4K Assembler/Editor..$85.00

Special Package Price ***$599.00***

4K AIM-65, 8K BASIC ROM, Power Supply, and Case.

ATARI 800

Don't Miss Out on Our Special Sale Price

At last there is a machine designed to give serious competition to Apple and Radio Shack. This computer contains many advanced technical features such as: built in RF modulator for use with a standard TV; 8K of internal RAM (expandable to 48K), 8K BASIC language included; extremely sharp high-resolution color graphics; and built-in peripheral I/O ports. Software is available in plug-in paks and cassette tapes, with many programs available in the areas of entertainment, education, and business/home management.

Available accessories include a printer, disk drives, game controller paddles, and memory expansion cartridges.

And JADE has the Atari 800 **in stock** at a special introductory price.

ATARI 800 ..... **$825.00**
810 DISK DRIVE **$575.00**
820 PRINTER **$550.00**
16K RAM EXPANSION KIT **$169.95**
CX30-04 PADDLE CONTROLLER **$19.95**
CX40-04 JOYSTICK CONTROLLER **$19.95**
ATARI 400 ..... **$599.95**

Jade Memory Expansion Kits for

TRS-80 APPLE EXIDY

Everything you need to add 16K of memory to your computer. Your kit comes neatly packaged with easy to follow instructions. In just minutes your computer is ready to tackle more advanced software.

$59.95

Special Price for

NOVATION CAT

Acoustic Coupler/Modem

$149.95

Let your computer talk to other computers!
Bell Systems 103 compatible
300 baud, answer or orginate
IOM-5200A (SALE PRICED) ... **$149.95**

Guaranteed Premium Quality

JADE DISKETTES

Magnificent Magnetic Media™

5¼" single sided, single density, box of 10
MMD-5110103 (SOFT SECTOR)..... **$29.95**
MMD-5111003 (10 SECTOR) **$29.95**
MMD-5111603 (16 SECTOR) **$29.95**
5¼" double sided, double density, box of 10
MMD-5220103 (SOFT SECTOR)..... **$39.95**
8" single sided, single density, box of 10
MMD-8110103 (SOFT SECTOR)..... **$34.95**
8" single sided, double density, box of 10
MMD-8120103 (SOFT SECTOR)..... **$55.95**
8" double sided, double density, box of 10
MMD-8220103 (SOFT SECTOR)..... **$57.95**

SD Systems

Z-80 STARTER

Complete Z-80 Microcomputer

On-board keyboard, display, EPROM programmer, and cassette interface
On-board S-100 interface
Wire-wrap area and room for 2 S-100 connectors
Two 8-bit parallel I/O ports, 4 channel CTC,
5 programmable breakpoints
Examine & change memory, I/O ports, or register
CPS-30010K (KIT) **$299.95**
CPS-30010A (A&T) **$369.95**

Latest Version, Special Price

WORDSTAR

Universally interactive text editor/word processing package. Very powerful and easy to use on all types of computer systems.

SFC-13600100F **$475.00**

PLACE ORDERS TOLL FREE

Inside California 800-262-1710
Continental U.S. 800-421-5500

For customer service or technical inquiries call 213-679-3317

Write for our FREE 1980 catalog

JADE COMPUTER PRODUCTS

4901 W. Rosecrans, Hawthorne, CA 90250

TERMS OF SALE: Cash, checks, credit cards money orders or from recognized institutions Purchase orders accepted. Minimum order $10.00. California residents add 6% sales tax. Minimum shipping and handling charge $2.50. Prices are for U.S. and Canadian delivery only and are subject to change without notice. For export prices and information send for a JADE INTERNATIONAL CATALOG.

VISA AMERICAN EXPRESS master charge

Circle 160 on inquiry card.

Peripherals available for the Atari 800 include a cassette recorder, printer, and 5-inch floppy-disk drive. The cassette recorder, which comes with the Atari 800 and is an extra-cost option on the 400, features a digital tape counter and computer control of the motor. However, the computer cannot make the cassette rewind or fast forward. You must press keys on the cassette machine to perform these functions.

Within these physical limits, you can do quite a lot with the cassette recorder besides saving your own BASIC programs for future use. Saving programs on cassettes enables you to build up a library of software. Or you can save data (personal financial information, for example) in cassette data files. A BASIC program may then input data from those files, operate on it in some way, then print out results or save the results in a second data file (if you have a second cassette recorder). The file system controlling the cassette unit lets you name your cassette files, making your library of programs and data files accessible in a reasonable way.

A forty-column-wide printer may also be connected to the Atari, so that programs and data can be printed, saved, or both.

Disk Drive

The most exciting thing about Atari's disk drive, which connects only to the Atari 800, not the 400, is not the hardware but the flexible and powerful *file system* that manages the hardware. The programmer need not worry about *track* and *sector numbers*. (Programs are kept on a series of concentric tracks, like bands on a phonograph record. Each track is subdivided into areas called *sectors*.) Programs, text, and data may all be stored in named files. You can read from or write to any file on a character basis, which makes files act just like any other input/output device—and you may have up to four files open at one time. Such file handling enables the development of a rich array of increasingly powerful *software tools* not only by Atari and other software professionals, but even by the untutored individual who teaches himself by writing programs in Atari BASIC.

Electromagnetic Interference

The Atari 800 has a very clean design right down to its internal metal shield designed to trap radio frequency interference (RFI). RFI results from the high frequencies at which a microprocessor operates. The Atari's microprocessor is a 6502, operating at a crystal-controlled clock frequency of better than 1 megahertz, or one million cycles per second. This means that logic gates switch and currents change every millionth of a second.

Every time the current changes, it generates an electromagnetic ripple. This is true of any electrical current switching on or off. Turn on a lamp while you're watching television. Bip! There's a ripple on the screen. Turn the current on and off a million times a second and you'll generate radio noise in exactly the same frequencies used for broadcast television. So a personal computer might not only generate an intended display for you, but unintended interference for your TV-watching neighbor. [*Turn on an FM radio near many of today's personal computers while they're working, and you'll hear the effects of RFI....***ed**]

Nobody likes a noisy neighbor, and in an electronic culture, radio noise is as unneighborly as a barking dog. The FCC requires manufacturers of personal computers to certify that their products do not exceed allowed levels of electromagnetic interference. Atari has been designed to meet these standards, and you can see how when you lift the lid to the program cartridge compartment. Although the Atari 800 has a body made of high-impact plastic, another case made of metal is inside it.

The logic and memory hardware is all inside this metal box, or *Faraday cage*. It is virtually seamless and designed to trap radio noise emanating from the Atari's circuitry by grounding it before it can escape into the environment. That metal box looks very solid, as if it were cast for industrial use. The whole computer feels solid, tight, sealed.

The Living Room Computer

The Atari 800 is so well packaged that it is the first personal computer I've used that I'm willing to set up in the living room. My other computers look like a rat's nest of cables, metal, and handwritten labels indicating where all the cables should go: I would rather disconnect a telephone than disconnect one of these fragile brains. But the Atari looks like a little plastic typewriter; a showpiece computer. And, if necessary, you can disconnect it and set it up someplace else in a minute or two.

Many little features of the Atari computer make good sense. For example, the SYSTEM RESET key, which enables you to interrupt a program in progress and restart the computer, is well away from the typewriter keys, at the top of a column of special function keys. You're not likely to hit SYSTEM RESET unintentionally. Some other

systems put the equivalent key, which is potentially the most destructive key on the keyboard, right next to a frequently used key (usually the RETURN key). Atari put SYSTEM RESET out of the way and put high plastic ridges above it and below it. Now it's almost impossible for a stray keystroke to clobber a program by initiating a system reset. I applaud Atari for protecting me from my own stray keystrokes.

Documentation

The hardware makes the Atari work, and the software makes it work for the user; but Atari's documentation makes it the first personal computer I've used that I could unreservedly recommend to someone who knows nothing about computers. The Atari documentation tells and shows, with many pictures, how to set up the system, how to use a program cartridge, and how to write, save, and use programs in BASIC. The documentation even includes a book on programming in BASIC, as well as an explicitly detailed, step-by-step manual of Atari BASIC for those new to programming.

Also, the Atari documentation is not just for the beginner who wants to learn BASIC. An assembly-language programmer should find the Atari 800 (and presumably the 400) a joy to program, because so many operating-system subroutines and addresses are well-documented. Other systems may have operating-system software that is just as much fun to call directly, but if the software isn't documented, you can't use it very easily. Some manufacturers of personal computers have spawned a cottage industry by neglecting to document their operating systems: some programmers who buy such computers decode the operating systems themselves by *disassembling* the contents of the read-only memory, and then publish their findings for other owners hungry for such documentation.

Such a cottage industry is not likely to grow up around the Atari. Not only does Atari publish the addresses of subroutines and data areas that the advanced programmer might wish to access, but they even show the beginner how to do sophisticated systems programming . . . in BASIC! If you know a little systems programming, imagine how much more you could do with that hardware, and with all the well-documented software for that hardware!

So the Atari is more than a video game, and it's more than a bare-bones computer. It's a fully supported computer *system*. And a very fine system at that.■

About the Author

Ken Skier writes software for Wang Labs and teaches creative writing at Massachusetts Institute of Technology. His book Beyond Games *will be published by BYTE Books in 1980.*

THE SOURCE

Big Computer Power in Your Home

by Ken Skier

One day last May my friend Mike, a computer designer, told me some exciting news.

"There's a time-sharing system available for personal computer users," he said. "All you need is a telephone and terminal. You can run BASIC and FORTRAN programs and tie into a real-time data base—which even includes the UPI wire service! And you can search the data base by keywords. It's the future, Ken. Here! And you can get it for $2.75 an hour!"

For the past few weeks I have had access to that system, The Source, under circumstances that have been for me like giving a kid the key to a candy store. onComputing supplied me with a terminal and an account number for The Source, and told me to use as much as I liked. I have. Also I have observed a dozen other people using The Source. The printouts from these sessions, and the reactions of those users, have contributed greatly to this review.

What Is The Source?

Billing itself as "The Information Utility," The Source is a telecomputing network that gives the user "access to literally thousands of programs and data bases, including the ability to communicate electronically with other...users" interactively and through electronic mail. Users may enter and run programs in BASIC, FORTRAN, and COBOL, or may use programs that have already been written and tested and which are available under any of a number of public program libraries. Perhaps the most exciting operation is to access real-time news, sports, financial, travel, and consumer data bases. The Source is a service offered by the Source Telecomputing Corporation (STC).

Signing On

In order to use The Source, you turn on your terminal, turn on your *modem* (a device that lets your computer talk to other computers over the telephone) and pick up your telephone. Then you dial the nearest number in the Telenet Telephone Access list that comes with your Source documentation.

What comes next is trickier. You must put the handset on the modem, and conduct a cryptic dialogue with an automated interrogator at the other end. This dialogue is reminiscent of those great street-corner scenes from spy movies: sign/countersign/counter-countersign/counter-counter-countersign...and *then* the guy in the trenchcoat gets the microfilm.

Well, spy movies are camp, and I know The Source needs to know that I'm a subscriber before it lets me loose on the system, but I wish I didn't feel like such a gambler every time I try to sign on. Out of my first six attempts at using the system, only three ended with me on-line. Sometimes The Source didn't seem to be there ("301 24 NOT OPERATING"); sometimes it seemed too busy to deal with me ("301 24 BUSY"); and sometimes it just didn't like my countersign ("ID NOT VALID"). Things weren't made easier by the fact that my documentation contained two different sign-on procedures. After a while I found the one that worked, but even that didn't insure against my receiving such discouraging

responses as "301 24 NOT OPERATING" and "301 24 BUSY."

Once you're on The Source, what can you do? In the world's biggest candy store, which aisle do you explore first? The first time I logged onto The Source, I knew I could find out the closing prices on the New York Stock Exchange, calculate the bending moment in a structural beam, or write a program to perform some socially-useful function. I decided to play games.

Games

I had heard for some time about the great games you could play on time-shared computer systems, and had always felt left out. (The only computer games I ever played I had to key laboriously into my KIM-1 computer, and they were so simple that they never held my interest for more than a few minutes.) But here on The Source, I could play any of a number of games, and I wouldn't have to write them or type them into the system: they were already running, and available on-line. At last I could play Adventure, Star Trek, Hammurabi!

There is not sufficient space here to review these games in detail. But, briefly: Adventure *is* all it is supposed to be—a giant network of caves filled with treasures and tools and hazards, including dragons and knife-throwing dwarves. Adventure on The Source is the original Adventure developed by Don Woods, et al, at Stanford University. A complete game may take several hundred moves, but you don't need to complete a game in one session. When you wish to leave the terminal, you can save the game you're playing and resume it later from your current position in the caves. That's convenient. You can save your game and then ask some more experienced player how to get out of the desperate situation you've wandered into.

Star Trek is another story. It isn't nearly as "smart" as Adventure, which can understand dozens or hundreds of commands. Star Trek can only understand about a dozen commands, and if you don't give a command in precisely the proper format, the program responds with the very unhelpful prompt, "FOOL, THAT IS AN ILLEGAL COMMAND." Make the mistake of firing a photon torpedo at the wrong angle and it will respectfully tender the following report:

> "TORPEDO MISSED. USE YOUR COMPUTER TO CALCULATE THE PHOTON TORPEDO COURSE, STUPID."

Hammurabi is a lot of fun to play. As "Ruler of Ancient Sumeria," you must choose each year how much grain to feed your people, how many acres of land to buy or sell, and how many acres to plant with seed. Mismanage the economy and some of your people will starve; starve enough of your people and they'll ride you out of town on a rail. I found one minor bug in the program, but for the most part Hammurabi is an entertaining, educational experience. The program gives you no guidance—at first you have to guess how much grain people need to keep from starving—but it lets you learn quickly by trial and error. The first few times I played I was ridden out of town on a rail by a starving, outraged population, but more recently my rule has been compared to "the best years of Disraeli." I wish there were more such interactive simulations available on The Source. They are an ideal way to introduce new users to the power of the computer.

The Source provides users with access to the United Press International (UPI) wire service, and lets you search the data base of current stories by keywords, either individually or in combination. Thus, you can ask for all of the stories on CARTER, or all of the stories on CARTER and ENERGY, or all of the stories on COAL or OIL or INFLATION.

Data Bases

One of the most exciting features of The Source is the group of real-time data bases it makes accessible to subscribers. These data bases include information on virtually all American and many foreign markets for stocks, bonds, gold, silver, commodities, etc. Many people will want to use this data. But what I find most attractive is not the information on monetary matters, but the many other news stories that never quite find their way into my newspaper. (And rightly so; who wants a twenty-pound newspaper?)

The Source provides users with access to the United Press International (UPI) wire service, and lets you search the data base of current stories by keywords, either individually or in combination. Thus, you can ask for all of the stories on CARTER, or all of the stories on CARTER *and* ENERGY, or all of the stories on COAL *or* OIL *or* INFLATION. Stories are kept in the data base until they are six days old; only then are they purged from the system. That gives you a real-time data base of everything the UPI provides to newspapers.

Pretty exciting, but not easy to use! The first time I entered the UPI data base, I had no trouble finding and reading stories filed from the site of the antinuclear demonstration at Seabrook, New Hampshire. That was fun. I had a newswire Teletype chugging away in my own home, and I could rip out the latest story and read it, just as hundreds of practitioners of the

"Rip and Read" school of journalism do over the air every day.

But then a friend visiting from Denver wondered about the weather back home. I said, "Why wonder? We can find out. We'll just ask The Source."

The manual suggested using the command DATA WEATHR. We did, and the system advised us to use the command DATA UPI, and then query the UPI data base for the weather in selected cities by typing, "PM-TEMPS." So we tried DATA UPI, to enter the UPI data base, and were immediately deluged with information on how to access particular stories. But when the torrent of stories and story numbers subsided and we typed in "PM-TEMPS," the system didn't respond with the weather in selected cities. Rather it told us we'd issued an illegal command. Yet it was the command The Source itself had told us to use! So we tried to get out of the UPI data base by using the BREAK key, and that didn't help. In fact, none of the ways to escape from other programs seemed to get us out of the UPI data base. Finally we hung up the phone.

Since then I have learned how to use the UPI data base, and in fact I can get the weather report any time I want. The data base consists of many levels, and certain commands are effective only at certain levels. To jump up a level within this structure, you simply type "quit"; by typing "quit" repeatedly, you keep popping up a level until you are out of the data base entirely, and are back in system-level command mode.

Unfortunately, I did not learn about this structure from the documentation, but rather from a friend who has worked for years with time-shared data bases, and for a summer with UPI. And even he had to figure it out by playing with The Source for an evening. The UPI data base has a simple but powerful structure, but that does not help the user who doesn't know how it works. The information is there, but The Source documentation doesn't always tell you how to ask for it.

When you buy software for your stand-alone microcomputer, you are pretty much on your own. But when you use the software on a time-shared system, the system is obligated to support it.

The Source is very powerful compared to any stand-alone home computer, but its documentation is inadequate and in some cases inaccurate.

Standards

There is more involved here than documentation. A computer system is not a system simply because all of the parts of it are wired together: every part of it can (or should) be expected to behave in the same way. A system implies standards, conventions, and a predictable adherence to those conventions.

Some of The Source's commands behave consistently from Program A to Program B, but I have not found any commands yet that behave consistently across all programs. Just when I think I have found a standard response to a given keystroke or command, I encounter a program with a different response to that command or keystroke. When the people at STC impose and maintain true command standards, the extensive program library of The Source will become accessible to even the naive user, who will know that no matter where you may go while in the system, you can always get help by typing "HELP", and you can always exit gracefully by hitting the BREAK key. Knowing that you can get out again, you may be much more likely to go in.

Program Libraries

After you have played enough games and read enough news, you might wish to explore the program libraries available on The Source. These libraries include programs for financial analysis and for scientific and engineering problem-solving. However, my financial affairs do not yet require the use of a computer, and I don't often find myself figuring the bending moments in beams, so I am not in a position to evaluate these programs. Presumably they are well written and will be maintained—which, if true, should make these programs more dependable and more useful than programs written to perform similar functions on stand-alone systems. Why? Because programs written for stand-alone systems might not be supported after they are sold, and the people who provided those programs may no longer be around to answer your questions or (dare I say it?) to provide you with source listings so you can modify them yourself. When you buy software for your stand-alone microcomputer, you are pretty much on your own. But when you use the software on a time-shared system, the system is obligated to support it. More importantly, the system has a direct and continuing monetary motivation to support its software, if it expects to keep charging you for connect time.

Furthermore, programs which require large files and buffers may

be very difficult to implement on stand-alone microcomputers, whereas large files and buffers are likely to fit effortlessly into the large computer systems on which The Source runs.

On-Line Storage

Speaking of programs that fit effortlessly into available memory, let me mention here that you may store as many files as you want on The Source, and each file may be as long as you like. This means that you can store lots of different programs, or lots of versions of similar programs, as well as large files of data and text. And you don't have to worry about where they are; they are all on-line. Never again will you have to locate a cassette or a disk that you know has just what you want on it. When you are on The Source, you simply ask for your file by name, and you have it! What an intoxicating opportunity for programmers who have previously watched every byte, because there just was not any more storage than the limited capacity of their own systems. With The Source, you have unlimited storage available at a cost of literally pennies per month. What a resource!

The Text Editor

The most important piece of software on any computer system is the text editor. A text editor is a program that lets you enter text into a file, and then review, edit, and/or print that file. For example, you may create a file that consists of the single phrase, "The quick foxes jumped over the dog." Later you may go back and insert "brown" and "lazy," and the text editor will then show you the new version of your file: "The quick brown foxes jumped over the lazy dog."

When you code a program in any language available on The Source, you must use a text editor to create the source program file. You might even want to use the text editor to create a file that is not a program: for example, a letter, a term paper, a report, or a proposal. The Source claims that the text editor can be used to create "letter-perfect correspondence." And it can, but not very easily. And certainly not very quickly.

The Source text editor is a line-oriented editor, which means that it considers each file to be a set of lines, rather than a very long string of characters or a set of sentences or pages. Lines are reasonable units for program files, but not for documents written in English. The text editor doesn't "think" about text the way the human user does, so the user must try to adapt to the way the editor handles text. It's not easy.

Even to correct a typographical error can be more trouble than it is worth. The text editor recognizes the RUBOUT character, which my terminal can send, and the documentation says to strike the RUBOUT key when you wish to delete a character. Thus, if you type "Good morninr" and realize that you've just hit the wrong key, you need only strike RUBOUT to cause the text editor to delete the "r" from the end of that line. Unfortunately, the text editor can take as long as ten to fifteen seconds to respond to that keystroke. While you're waiting, you have no idea whether or not the keystroke was registered at the other end, so you might hit RUBOUT again, just to be sure. That would be a mistake. When the text editor does get around to registering your first RUBOUT, it will not stop there, but will then register your second RUBOUT, leaving you with more characters deleted than you had intended to delete. Sometimes the text editor responds to characters such as the RUBOUT within a second, but quite often the delay is longer than ten seconds, which I consider intolerable. A text editor is supposed to be superior to a typewriter, but I can correct a mistake on a manual typewriter in a lot less than ten seconds ... and when I've done it, I know I've done it.

Responsiveness

The Source is inefficient with regard to my time. This inefficiency is manifested in large ways, such as when I am forced to sit through quite a lot of information on the UPI data base, even when I've learned how to use that data base; and it shows up in seemingly small things, such as the often interminable delay in echoing RUBOUTs to the user's terminal. The Source does not seem to have made any effort to provide the user with the power to move quickly through informative messages and to zero in on the information the user seeks.

The Source unfortunately presents its informative messages in a sequential manner, and if the message is ten pages long, you might have to read ten pages to find the information you want. The only control you have is to acknowledge after the screen has been filled with data that you are ready to read another screen. The Source should be able to *search* a long message for specific information. Thus, instead of having to read a list of every syndicated feature carried in the UPI data base, simply to get the story

number for a given columnist, you could have the system search that list for the name of the desired columnist. For example, if I want to read Art Buchwald's latest column, I do not want to see a list several screens long of every other feature they've got.

Cost

A subscription to The Source has an initial cost of $100, which provides the user with an account, an account number, and the basic set of documentation. A subscriber may then use The Source at any time, paying *connect-time* charges of $2.75 per hour from 6 PM to 7 AM, and $15.00 per hour from 7 AM to 6 PM. On-line storage costs about a penny per 2000 characters per month.

You must pay for the cost of the phone call to the nearest Telenet telephone line. There are Telenet numbers in most major cities, so if you live in a major city this call will probably be toll-free for you. If you live outside of the local area for the nearest Telenet number, though, you will have to pay for the call. Depending on your location and your phone service, that could be a considerable expense, perhaps exceeding the direct cost of connect time itself. Furthermore, if you plan to use The Source very often, the way people use a television or a newspaper, then you may wish to install a second telephone line, or you'll be incommunicado every time you delve into the UPI data base or play Hammurabi.

The services discussed in this review, as well as many others, are available to subscribers at no cost beyond that of connect time. Other services require the subscriber to pay a fee over and above the cost of connect time. For example, the user may arrange to have files printed out by The Source and then mailed to the user, or to another recipient if the user specifies.

Certain programs require the

article policy

onComputing is continually seeking high-quality manuscripts about personal computing. Possible topics include explanatory articles about the workings of computers, how to program them, and how to purchase them. Articles about applications of personal computers to everyday life will also be considered, as well as lucid accounts of more technical applications. Our audience includes talented lay people as well as those more versed in computer lore; accordingly, all technical terms in articles should be defined, and all acronyms spelled out. Manuscript pages should be typewritten and double spaced. For a more formal description of procedures and requirements, potential authors should send a large (9 by 11 inch, 30.5 by 22.8 cm), self-addressed envelope, with 28 cents US postage affixed, to onComputing Author's Guide, 70 Main St, Peterborough NH 03458.

user to pay a special fee. Part of this fee is a royalty to the program author. Thus, if you write a program that you think other people may wish to use, you can submit it to The Source for inclusion in a public library. If The Source includes your program in a public library, then every time someone uses that program, you will get the amount of your royalty credited to your account. You might make enough from royalties to pay completely for your use of The Source.

It is possible to rent a terminal and modem from The Source, or you may rent elsewhere, or buy or build your own. Check the ads in *BYTE* magazine to see what a terminal and modem cost, both assembled and in kit form. If you have your own personal computer, you can almost certainly get a program for it that will make it behave like a computer terminal; then all you'll need will be a modem. Check with your local computer store for more information.

Without doubt it costs less to subscribe to The Source than to own your own computer system. Rule Number One of Personal Computing says that if you buy your own computer system, it will never be complete. Your system will always be crying out for a disk drive, or a printer, a graphics display, or more memory. (It *always* wants more memory.) But when you subscribe to The Source, you won't be likely to feel those temptations: the system is there, just beyond the phone line, it's tremendous, and there's very little you can do to extend it. On the other hand, if you want to get very "close to the bytes" (ie: to become computer-literate by learning one machine very well), then you should probably get the best personal computer you can afford. A small computer is one that you can really get to know, from keyboard to display. And speaking of displays: if you want to develop and/or use screen-based interactive programs such as animation and word processors, then The Source is ill-suited to your needs, since you can't easily do screen manipulations over a telephone line. For such applications you should get a personal computer with a video display *and* the software to support your application.

When you are on The Source, you simply ask for your file by name, and you have it! What an intoxicating opportunity for programmers who have previously watched every byte, because there just was not any more storage than the limited capacity of their own systems.

But if you don't feel ready to spend several hundred to a few thousand dollars on your own computer system, you can spend a few hundred and get to know The Source. When you think that your few hundred dollars will enable you to program in several computer languages (not just one or two, which is all you can get on most home computers), and furthermore will give you access to several significant data bases, you can see why the people at STC think they'll reach a very large market this year.

Conclusion

Its massive memory and data bases make The Source more powerful than any stand-alone home computer system, and yet it is less responsive in many ways than such a small system can be. It provides more information in its data bases than anything else available to the ordinary consumer, and yet that information is not very easy to peruse or retrieve.

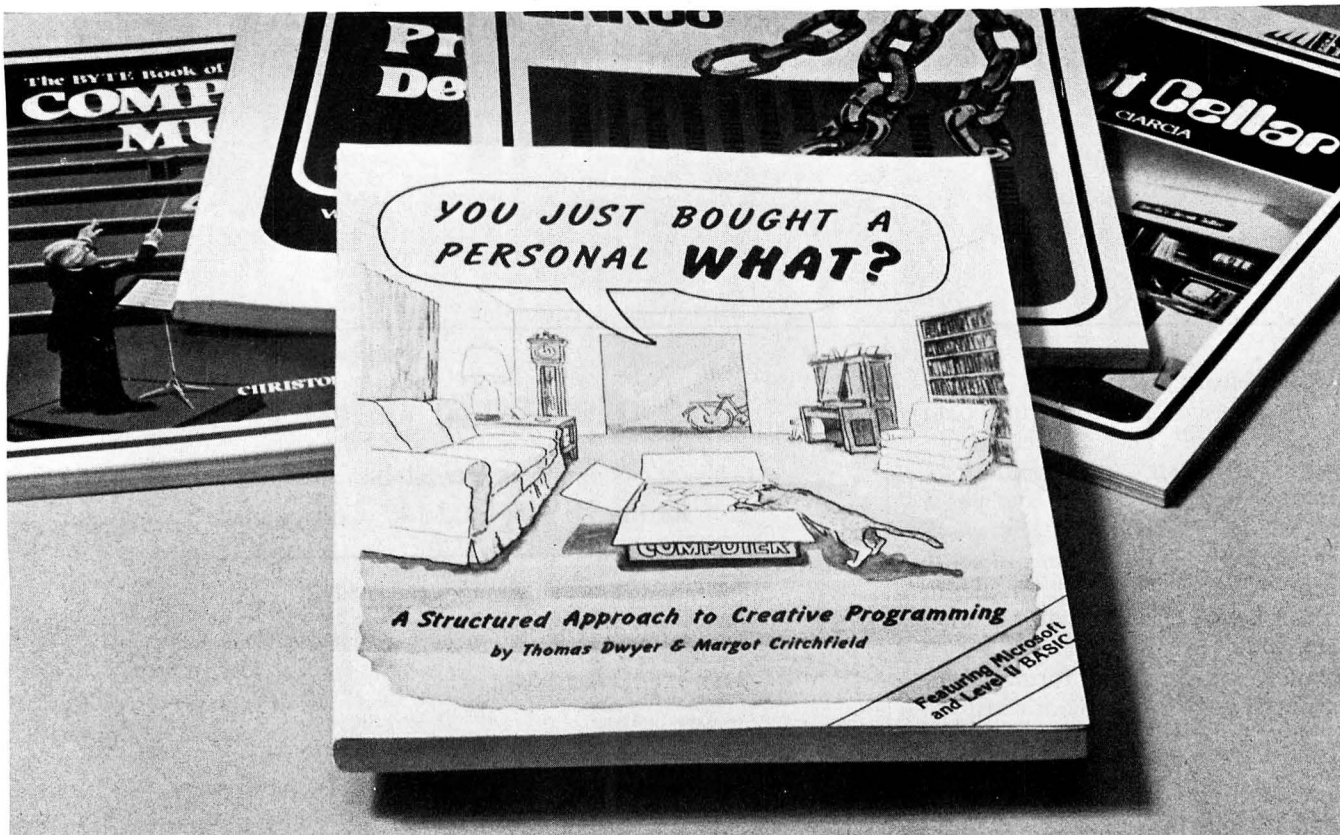
But there is good news. Very good news. All of these problems can go away. As The Source reaches out to thousands of first-time computer users, it will adapt. It will have to. It will move to a standard and intuitively reasonable set of conventions, and will cease to include software that locks the user in loops. It will develop easier-to-use and more powerful text editors ... perhaps a whole series of text editors, each designed to take advantage of the features available on a given type of terminal. (An editor that could drive terminals with X,Y cursor positioning, for example, might make real-time, on-screen editing available to anyone with the right kind of terminal. What a wonderful thing that would be!) And the UPI data base, which alone will motivate many people to subscribe to The Source, will of necessity become easier to use.

As more users enter the system, and more programmers write software, the main contribution of STC will be to manage that software, presumably with standards imposed before the software is released. Manage the software and maintain it, and make it available to as many people as possible: that's a big job for a new organization. (Source Telecomputing Corporation has been in existence only since last June.) But it can result in incredible power being made available to almost anyone. No longer do you need to be a programmer, a corporate executive, an accountant, an engineer, a journalist, a banker, a meteorologist, or a student, to have access to information, information processing, and text processing.

Just as bigger and better libraries offer more and better resources to meet the needs of a community of readers, so will bigger and better computer networks and data bases offer more to meet the needs of a community of computer-literates.

The Source is the first available example of a concept that is sure to be with us for a long while.■

For more information about The Source write Source Telecomputing Corporation, 1616 Anderson Rd, McLean VA 22102.



You Just Bought a Personal What?

by Thomas Dwyer and Margot Critchfield

Whether you are a novice programmer or an experienced computer user, this book is filled with practical ideas for using a personal computer at home or work. It will take you through the steps necessary to write your own computer programs, and then show you how to use structured design techniques to tackle a variety of larger projects. The book contains over 60 ready-to-use programs written in Microsoft and Level II BASIC in the areas of educational games, financial record keeping, business transactions, disk-based data file and word processing. **$11.95 pp. 256 ISBN 0-07-018492-5**

Please send ☐ _____ copies of **You Just Bought a Personal What?**

Name Title Company

Street City State/Province Code

☐ Check enclosed in the amount of $_____

☐ Bill Visa ☐ Bill Master Charge

Card No. _____ Exp. Date _____

Add 75¢ per book to cover postage and handling.

BYTE BOOKS
70 Main Street
Peterborough, New Hampshire 03458

McGraw-Hill

A DO-IT-YOURSELF TEXT EDITOR

by T. E. LeVere

This simple line editor lets you edit text on your TRS-80 Level II or Heathkit H11 personal computer.

One of the most convincing arguments for personal computers is that they are often able to make life easier. One useful job for your computer is the chore of text editing. (See the word-processing articles in this issue.)

In theory, text editing is a simple procedure involving three tasks. First, the text material is entered into the editing system and displayed for review. This typically involves some kind of keyboard and video display. Second, the text is modified to the user's satisfaction. This involves the addition or deletion of material to make the text more readable and grammatically correct. Finally, the edited version is printed as hard copy.

While this seems simple enough, there are a few steps that complicate matters. First, there is an additional task to be accomplished if the editor is to be truly useful. This task is to "update" or organize the text material once some amount of editing has been accomplished. Making deletions and insertions during editing often produces a fragmented text, and it becomes necessary to clean things up so that the text is composed of complete, corrected lines of relatively uniform length.

In *dedicated* editors, this is usually accomplished each and every time the text is modified. For the smaller general-purpose computer, this slows the editing process down to a rate tolerable to only the most patient of users. Thus, more often than not, there is considerable advantage to letting the text break up during editing. Then, when the process gets to the point where the text is difficult to read, the user instructs the system to organize the material into a more readable form. If editing is to be accomplished with any ease and elegance, the system must be both line- and character-oriented. This allows the user to examine a particular line of text and to change a particular character within that line. Also, the system should do this both "forward" and "backward"; that is, text at any location can be accessed, regardless of the present display.

Finally, and most important for generality, the editor must be written in a language easily adaptable to small, general-purpose computers.

I have chosen to write my word processor in BASIC. BASIC runs a distant second to assembly language in terms of efficiency and flexibility. However, this is a small price to pay when compared with

the problems involved in making the assembly-language program of one microprocessor palatable to other microprocessors. Even accepting the shortcomings of an editor encoded in BASIC, there still remain some hardware considerations. In particular, the line- and character-oriented features of a versatile text editor necessitate the use of a *virtual character file*. (See the glossary at the end of this article.)

Virtual files enable the user to read or write text at any point in the file. This is a severe strain on common cassette systems, so that either floppy disks or large amounts of memory are necessary. Fortunately, disks are rapidly replacing cassettes in personal computers. However, if your computer system has only cassette-tape mass storage, I suggest you read Fred Ruckdeschel's excellent article, "BASIC Text Editor" (June 1979 *BYTE*, page 156), which describes a cassette-based text editing program.

On the other hand, if you have a disk system and a BASIC interpreter with some elementary string-handling routines, then what follows may be of interest.

EDITOR

I wrote the EDITOR program (presented in listing 1) using a Heath H11 computer with 24 K bytes or 12 K words of programmable memory, an H9 video terminal, a W27 floppy-disk drive, and an H14 line printer. The program is written in Digital Equipment Corporation (DEC) BASIC, which is the version that is supplied with the W27 disk operating system. [*A similar version of this program, written for the Radio Shack TRS-80 Model I with disk drive and Disk BASIC is illustrated in listing*

Listing 1: *The EDITOR program, written in Digital Equipment Corporation (DEC) BASIC for the Heathkit H11 personal computer. See listing 2 for a TRS-80 version.*

```
1       REM EDITOR 11-02-79
5       DIM I$(10)
10      OPEN "TEXT" AS FILE VF1$(500)=128
20      PRINT "OLD OR NEW TEXT"; \ INPUT R$
25      IF R$="OLD" THEN 35 \ IF R$<>"NEW" THEN 20
30      VF1(0)=STR$(0)
35      P=VAL (VF1(0))
40      W=68
41      PRINT "LINE LENGTH="; W
42      PRINT "CHANGE"; \ INPUT W$ \ IF LEN(W$)=0 THEN 44
43      W=VAL(W$) \ GO TO 41
44      PRINT "LINE LENGTH SET TO";W
55      PRINT "ACTION"; \ INPUT R$
60      IF R$="A" THEN 100
61      IF R$="I" THEN 200
62      IF R$="L" THEN 300
63      IF R$="C" THEN 400
64      IF R$="E" THEN 500
65      IF R$="U" THEN 600
66      IF R$="H" THEN 800
67      IF R$="Q" THEN 1000
70      GO TO 55
100     REM APPEND
105     IF P=0 THEN 115
110     PRINT P;TRM$(VF1(P))
115     PRINT P+1; \ INPUT R$
120     IF LEN(R$)=1 THEN 60
125     IF LEN(R$)<=W THEN 130 \ PRINT "OVFL" \ GO TO 115
130     P=P+1 \ VF1(P)=R$
135     GO TO 115
200     REM INSERT
205     PRINT "INSERT IN LINE"; \ INPUT L
206     IF L>P THEN 205 \ PRINT TRM$(VF1(L))
210     PRINT "BEFORE"; \ INPUT R$ \ B=1
215     N=POS(VF1(L),R$,B) \ IF N=0 THEN 55
220     F$=SEG$(VF1(L),1,N-1)
225     S$=SEG$(VF1(L),N,LEN(VF1(L)))
230     PRINT F$; \ INPUT I$ \ IF LEN(I$)>0 THEN 235
        \ B=N+1 \ GO TO 215
235     IF LEN(F$&I$)<=W THEN 240 \ PRINT "OVFL" \ GO TO
        230
240     VF1(L)=F$&I$
245     Q=1 \ L1=L
250     L=L+1
251     PRINT L; \ INPUT R$
255     IF LEN(R$)>1 THEN 260 \ I$(Q)=S$ \ GO TO 270
260     IF LEN(R$)<=W THEN 265 \ PRINT "OVFL" \ GO TO 251
265     I$(Q)=R$ \ Q=Q+1 \ IF Q<10 THEN 250 \ Q=Q-1
270     FOR I=P TO L1+1 STEP -1 \ VF1(I+Q)=VF1(I) \ NEXT I
275     FOR I=1 TO Q \ VF1(L1+I)=I$(I) \ NEXT I
280     P=P+Q
285     IF LEN(R$)>1 THEN 245 \ GO TO 60
300     REM DELETE LINE
305     PRINT "DELETE LINES FROM, TO"; \ INPUT F,T
310     IF F>T THEN 55 \ IF F>P THEN 55 \ F1=F
315     IF T<P THEN 325
320     P=F-1 \ GO TO 55
325     FOR I=T+1 TO P \ VF1(F)=VF1(I)
330     F=F+1
340     NEXT I
345     P=P-((T-F1)+1)
```

Listing 1 continued on page 71

Listing 1 continued:

```
350     GO TO 55
400     REM DELETE CHARACTERS
405     PRINT "DELETE IN LINE "; \ INPUT L \ IF L>P THEN 55
415     PRINT L; TRM$(VF1(L))
420     PRINT "CHARACTERS "; \ INPUT R$
425     B=1
430     N=POS(VF1(L),R$,B) \ IF N=0 THEN 55
435     PRINT SEG$(VF1(L),1,N-1);"(";SEG$(VF1(L),N,N+(LEN
        (R$)-1));")";
436     INPUT D$
440     IF D$="YES" THEN 450 \ IF LEN(D$)<>0 THEN 435
445     B=N+1 \ GO TO 430
450     VF1(L)=SEG$(VF1(L),1,N-1)&SEG$(VF1(L),N+LEN(R$),
        LEN(VF1(L)))
455     GO TO 55
500     REM EXAMINE
505     PRINT "EXAMINE LINES FROM, TO "; \ INPUT F, T \ IF F>P
        THEN 55
510     IF F>T THEN 55 \ IF T<P THEN 515 \ T=P
515     N=0
520     FOR I=F TO T
525     PRINT I;TRM$(VF1(I))
530     N=N+1 \ IF N<10 THEN 540
535     N=0 \ INPUT R$ \ IF LEN(R$)>=1 THEN 60
540     NEXT I
545     PRINT "END" \ GO TO 55
600     REM UPDATE
605     PRINT "UPDATE LINES FROM,TO "; \ INPUT F,T
610     IF F>T THEN 55 \ IF F>P THEN 55 \ IF T<P THEN 615
        \ T=P
615     N=F
620     F$=TRM$(VF1(F))
625     IF POS(F$,".",LEN(F$)-1)>0 THEN 628
626     IF POS(F$,"!",LEN(F$)-1)>0 THEN 628
627     IF POS(F$,"?",LEN(F$)-1)=0 THEN 629
628     F$=F$&" "
629     F=F+1 \ IF F>T THEN 685
630     S$=TRM$(VF1(F))
635     X$=F$&" "&S$&" "
640     IF LEN(X$)>W THEN 650
645     F$=TRM$(X$) \ GO TO 625
650     Z=0
655     Z=POS(X$," ",Z+1) \ IF Z>W THEN 665
660     Y=Z-1 \ GO TO 655
665     VF1(N)=SEG$(X$,1,Y)
670     N=N+1
675     F$=TRM$(SEG$(X$,Y+2,LEN(X$)))
680     GO TO 625
685     VF1(N)=F$
690     F=N+1 \ GO TO 310
800     REM HARDCOPY
805     OPEN "LP:" FOR OUTPUT AS FILE #7
810     PRINT "OUTPUT LINES FROM,TO"; \ INPUT F,T
815     IF F>T THEN 55 \ IF F>P THEN 55 \ IF T<P THEN 820
        \ T=P
820     PRINT "LINE SPACING"; \ INPUT L
825     IF L*((T-F)+1)>=65 THEN 840
830     PRINT "CENTER TOP-BOTTOM"; \ INPUT R$
835     IF R$="YES" THEN 845 \ IF R$<>"NO" THEN 830
```

Listing 1 continued on page 72

2. A few details of the operation of this version are given in a nearby text box . . . . **CPM]**

There are four string-handling routines of this particular BASIC that are used extensively by EDITOR:

- LEN(X$) — returns the number of characters (including trailing blanks) in string X$.
- POS(X$,Y$,I) — returns the first occurrence of Y$ within the string X$ starting at position I.
- TRM$(X$) — returns the string X$ with trailing blanks removed.
- SEG$(X$,I,J) — returns the string of characters in positions I thru J in string X$.

Additionally, DEC BASIC allows multiple statements per line, which reduces the program size and saves memory. The listing of EDITOR given in listing 1 makes use of this capability, mainly to group together logically related BASIC commands. When reading this listing of EDITOR, the individual statements are separated by a backslash (\ ).

EDITOR recognizes exactly eight commands:

A—append text to the end of the text file;

I—insert text within an existing text line;

L—delete existing lines from the text file;

C—delete characters from within an existing line;

E—examine all or part of the text file (similar to LIST in BASIC);

U—update the text file (remove short lines created by text insertion);

H—produce a hard-copy listing of the text file;

Q—close the text file and quit.

Listing continued:

```
840     PRINT "TOP MARGIN "; \ INPUT J \ GO TO 850
845     J = ABS((65 - (L*(T - F) + 1))/2)
850     IF W<80 THEN 860
855     K = 0 \ GO TO 900
860     PRINT "CENTER LEFT-RIGHT"; \ INPUT R$
865     IF R$ = "YES" THEN 880 \ IF R$< >"NO" THEN 860
870     PRINT "LEFT MARGIN "; \ INPUT K
875     IF K + W< =80 THEN 900 \ PRINT "OVFL" \ GO TO 870
880     K = (80 - W)/2
900     FOR I = 1 TO J \ PRINT #7: \ NEXT I
905     FOR I = F TO T
910     FOR H = 1 TO K \ PRINT #7:" "; \ NEXT H
915     PRINT #7:TRM$(VF1(I))
920     FOR H = 2 TO L \ PRINT #7: \ NEXT H
925     NEXT I
930     CLOSE #7 \ GO TO 55
1000    VF1(0) = STR$(P)
1005    CLOSE
1010    END
```

The remainder of this article will explain each of these commands in the order given.

Each command operates on part or all of the text file as defined by a range of line numbers requested by EDITOR immediately after the command is initiated. While this, of course, introduces some duplication of program lines that will make the purist cringe, it does have an advantage. It allows individual users to easily modify a command of EDITOR to suit their own requirements without destroying the logic of the remaining portions of the program.

Understanding the Program

Before considering the individual commands, a few words concerning the general structure of EDITOR and its initial start-up are necessary. Basically, EDITOR operates on a virtual file called TEXT and is able to accommodate 500 character strings (lines) of 128 characters each. This file is *opened* as VF1 at program line 10. Line 10 must be modified for systems other than the H11. Independent of system particulars, the file is simply a list (an array of one subscript) of the general form VF1(X), where X is any integer or integer variable within the range of 0 thru 500. This means that any line of the list can be displayed or changed by specifying its number, or position in the file.

When EDITOR is started, the user is given the option of processing OLD or NEW text. The first string in the file, VF1(0), is always loaded with the number of strings in the file. If OLD is entered for example, EDITOR uses the information in VF1(0) to set the pointer P to the last line of text. If NEW is entered, P is set to 0. Throughout the program, P points to the last non-empty line of text in the text file. When the user ends the current command, VF1(0) is again set to indicate the last line of text.

Line 40 sets the value of variable W, which represents the number of characters per text line. The width of 68 characters per line was dictated by my H9 video terminal, which allows a maximum of 73 characters. A 3-digit line number, a space, and the BASIC prompt character "?" take up a total of 5 characters, leaving 68 characters for a line of user text.

Nevertheless, EDITOR asks whether the user wishes to change the line length (line 42 in listing 1). Entering a number at this point sets W to that value. Simply hitting the return key leaves W at its preset value. EDITOR then displays the line length to be used for this run of the program. (See listing 3 for an example of this dialog.) The line width W may be changed in order to modify the width of the printed output. I have found that the ability to change W is useful in several situations.

Commands

When execution of the program reaches line 55, EDITOR asks what action the user wishes to take (see listing 3). The only response EDITOR will accept is one of the valid commands: A, I, L, C, E, U, H, or Q. If any other letter, number, or string of letters and numbers is typed, EDITOR simply ignores it, and repeats the request for a valid command. When the user types a valid command, the program will then branch to the appropriate line group and execute the command under the user's direction.

Input Commands: A and I

Let us assume that the user wishes to edit new text material. After informing EDITOR that this is the case and after setting the appropriate line length, the user types A and hits the return key. EDITOR immediately branches to the 100-line group. The statements in this section cause EDITOR to display the line number and text of the last line in VF1; EDITOR then displays the next succeeding line number, the BASIC prompt "?" and waits. The prompt indicates that EDITOR is waiting for the user to enter material from the keyboard. This information will then be associated with the displayed line number.

There is one restriction, however, that the user must observe: the text string must be greater than a single character, or EDITOR will interpret it as a

The TRS-80 Version of EDITOR

The version of EDITOR given in listing 2 is somewhat different from that of listing 1. Most of the differences are due to the inevitable incompatibilities between two companies' versions of BASIC. The STR$, POS$, and TRM$ functions of DEC BASIC have been replaced by Radio Shack Disk BASIC equivalents, MID$, INSTR$, and two subroutines at lines 2000 and 2020 in listing 2.

The interaction between the EDITOR program and the disk file is handled differently for the Radio Shack disk system. The variable VF1$ is shared by BASIC and the disk file. It is *always* 128 characters long (see line 17), which necessitates the use of variables VQ$ and F$ and the occasional use of the phrase, LSET VF1$ = . Also, a random file in Disk BASIC has exactly 335 records. Since one record is used to store the number of text lines in the file, a text file can have a maximum of 334 lines (as opposed to 500 lines for the Heath H11 computer). Since these lines are 255 characters long, some additional programming could enable the program to store two or even three lines per random record, thus increasing the effective size of the text file.

In the character-delete section (lines 405 thru 455), a response of NO can be used to exit the command without deleting anything in the line under examination.

In the hard-copy section (lines 806 thru 930), provision has been made to output to the video screen or the printer. Do not use the Y option (print to printer) if you do not have a printer attached to the Expansion Interface and turned on. The variable PW contains the length of the print line; it affects the margin width of the printed output if the user chooses to center the text horizontally.

Finally, if space is a problem in your TRS-80 disk system, the remark lines in listing 2 can be removed without affecting the operation of the program.

Our thanks go to the Radio Shack Associate Store in Keene NH for their help with this article.

single-letter command and branch back to evaluate the command. To circumvent this problem, simply add a space next to the single-character line so that EDITOR will accept it as a text string.

When a text string is typed, EDITOR checks to see if it is within the line-length limit. If it is not, the EDITOR displays the message OVFL and asks for a shorter line by displaying the line number and the BASIC prompt "?". When an acceptable line has been typed, P is incremented by 1 and VF1(P) is loaded with the text string just typed. Following this, the next line number is displayed along with the BASIC prompt "?" and EDITOR again waits. To exit the append mode, enter a single-letter command to be executed next.

The second input command, I, inserts characters in an existing line. (See Listing 4.) The lines associated with this command all have numbers in the 200 group. When this part of the program is executed, EDITOR displays the prompt INSERT IN LINE?, and the only response it will accept is the number of an existing line of text. Following this, EDITOR displays the text of the selected line, then displays the prompt BEFORE? It should be noted that, even though it may be redundant to display a line of text that may already be on the screen, the isolation of the selected line between EDITOR's two prompts makes operating on the line considerably easier.

When some unique combination of characters is typed in response to the prompt BEFORE?, EDITOR will display the line to that point. If there is more than one occurrence of these characters, EDITOR will stop after the first occurrence. If this is not the occurrence intended (or expected), simply hit return and EDITOR will continue to the next occurrence. If EDITOR does not find the set of characters by the end of the line, it branches back to the section of the program that asks for a new command.

When the proper point of insertion has been found, enter the text to be inserted and hit return. EDITOR then determines if the length of the line to the insert point plus the length of the inserted material is larger than the line length permits. If the line is too long, EDITOR reminds the user by typing OVFL, and the process starts again. If the line length is within limits, the inserted text is added to the displayed line; EDITOR then displays the next line number and waits.

At this point, the user may wish to add more text that will appear after the insertion and before the end of the old line. If not, the user simply types a single command character. EDITOR then automatically assigns the remainder of the line to the displayed line number, and the rest of the text is renumbered to accommodate the new line. Control is then transferred to the beginning of the program to evaluate the command character.

Finally, it should be noted that the insert command will always break a text line at the point of insertion; this is demonstrated in listing 4. As described below, this is easily corrected with the update command; but, the method used in the update procedure to combine lines makes it messy to insert characters within words. When EDITOR combines lines of text during an update procedure, it always adds a space between the last word of the first line and the first word of the second line. Thus, if the user attempted to correct the

misspelling of "EDTOR" by inserting an "I" before "TOR", the result following an update would be "EDI TOR". The delete character command C explained below could be used to eliminate the offending space; however, it is usually more efficient to delete the incorrect word in total, then insert the correct word. A subsequent update will then arrange the text lines and the words within the text lines.

Delete Commands: L and C

If the character L is typed in response to the prompt ACTION?, or if it is the only character typed after a line number, then EDITOR branches to the 300-line group. At this point, EDITOR displays the prompt DELETE LINES FROM, TO?. The user must then respond with two existing line numbers, separated by a comma. (See listing 5.) The range of line numbers is inclusive so that the first number is the first line to be deleted and the second number is the last line to be deleted.

However, there are three restrictions. First, both numbers must be positive. Second, the first number must be less than or equal to the second. Third, the first number must be less than or equal to the total number of text lines. The second number may be greater than the total number of text lines, in which case EDITOR simply deletes all text lines beginning with the first line number and continuing to the end of the text.

As an example, suppose EDITOR is working on 35 lines of text and the user wishes to delete all text after line 28. Typing either 29,35 or 29,99 would produce the desired result. On the other hand, if the user wanted to delete only line 29, then the command should

Listing 2: *The EDITOR program, modified to run on a Radio Shack TRS-80 Model I with one disk drive and Disk BASIC. (See the text box on the previous page.)*

```
1 REM  (TUTORIAL TEXT EDITOR)
2 REM  (CREATED BY T. E. LEVERE)
3 REM  (REVISED 11-02-79)
4 :
5 REM  (THIS SIMPLE TEXT EDITOR IS LINE ORIENTED AND STORES)
6 REM  (ONE LINE OF TEXT PER RECORD IN A RANDOM DISK FILE. )
7 REM  (RECORD 1 CONTAINS # OF RECORDS IN FILE--TEXT IS IN)
8 REM  (RECORDS 2-335.  THEREFORE, A TEXT FILE CAN HAVE ONLY)
9 REM  (334 LINES. )
10 :
11 REM  (VALID COMMANDS ARE:   A (APPEND)   I (INSERT)
12 REM  (  L(DELETE LINES)   C(DELETE CHARACTERS)   E (EXAMINE)
13 REM  (  U (UPDATE)   H (HARDCOPY)   Q (QUIT)   )
14 :
15 CLS:CLEAR 1000
17 OPEN "RANDOM",1,"TEXT"
19 FIELD 1,128 AS VF1$
20 PRINT "OLD OR NEW TEXT ";:INPUT R$
25 IF R$="OLD" THEN GET 1,1:VQ$=VF1$:GOTO 35
26 IF R$<>"NEW" THEN 20
30 LSET VF1$="   1 ":PUT 1,1
35 R$="":VQ$="":P=VAL(MID$(VF1$,1,5))
37 IF P=0 THEN P=1:LSET VF1$="   1 ":PUT 1,1
40 W=56
41 PRINT"LINE LENGTH = ";W
42 W$="":PRINT "CHANGE (OR <ENTER> TO LEAVE AS IS) ";:INPUT W$
43 IF W$<>"" THEN W=VAL(W$)
48 PRINT "LINE LENGTH SET TO ";W
55 PRINT"ACTION ";:INPUT R$
60 IF R$="A" THEN 105
61 IF R$="I" THEN 205
62 IF R$="L" THEN 305
63 IF R$="C" THEN 405
64 IF R$="E" THEN 505
65 IF R$="U" THEN 605
66 IF R$="H" THEN 806
67 IF R$="Q" THEN 1005
70 GOTO 55
99 :
100 REM  (APPEND--USE THIS COMMAND TO ADD NEW LINES TO THE END
101 REM  (OF THE TEXT FILE)
102 :
105 IF P=1 THEN 115
110 GET 1,P:VQ$=VF1$:GOSUB 2000
111 PRINT P-1;":";VQ$
115 R$="":PRINT P;":";:INPUT R$
120 IF LEN(R$)=1 THEN 140
125 IF LEN(R$)<=W THEN 130
126 PRINT "OVFL":GOTO 115
130 P=P+1:LSET VF1$=R$:PUT 1,P:R$="":VQ$=""
135 GOTO 115
140 GOSUB 2040:GOTO 60
199 :
200 REM  (INSERT--USED TO INSERT A WORD OR WORDS INTO AN
201 REM  (EXISTING LINE; HIT <ENTER> TO SEARCH FOR SAME STRING
202 REM  (LATER IN THE SAME LINE.  "INSERT" TAKES THE REST OF
203 REM  (THE LINE AND PUTS IT ON A NEW LINE)
204 :
205 PRINT"INSERT IN LINE ";:INPUT L:L=L+1
206 IF L>P THEN 205
207 GET 1,L:VQ$=VF1$:GOSUB 2000:PRINT L-1;":";VQ$
210 R$="":PRINT"BEFORE ";:INPUT R$:B=1
215 N=INSTR(B,VQ$,R$):IF N=0 THEN 55
220 F$=MID$(VQ$,1,N-1)
225 S$=MID$(VQ$,N,LEN(VQ$)-N+1)
```

Listing 2 continued on page 75

Listing continued:

```
230 I$="":PRINT L-1;":";F$;:INPUT I$:IF LEN(I$)>0 THEN 235
231 B=N+1:GOTO 215
235 IF LEN(F$+I$)<=W THEN 240
236 PRINT"OVFL":GOTO 230
240 LSET VF1$=F$+I$:PUT 1,L:F$="":VQ$=""
245 L=L+1
250 R$="":PRINT L-1;":";:INPUT R$
255 N=P:P=P+1
260 FOR I=L TO P
261 GET 1,N:PUT 1,N+1
262 N=N-1:NEXT I
265 IF LEN(R$)>1 THEN 275
270 LSET VF1$=S$:PUT 1,L
272 PRINT L-1;":";S$:GOTO 290
275 IF LEN(R$+S$)<=W THEN 280
276 PRINT "OVFL":GOTO 250
280 LSET VF1$=R$+S$:PUT 1,L
281 PRINT L-1;":";R$+S$
290 S$="":GOSUB 2040:GOTO 60
299 :
300 REM  (DELETE--THIS DELETES A LINE OR RANGE OF LINES, MOVES
301 REM  (THE REMAINING LINES TO FIT, AND UPDATES THE NUMBER-
302 REM  (OF-LINES RECORD IN RECORD 1)
303 :
305  PRINT"DELETE LINES FROM,TO ";:INPUT F,T
307 F=F+1:T=T+1
310 IF F>T THEN 55
311 IF F>P THEN 55
312 F1=F
315 IF T<P THEN 325
320 P=F-1:GOTO 347
325 FOR I=T+1 TO P
326 GET 1,I:PUT 1,F
330 F=F+1
340 NEXT I
345 P=P-((T-F1)+1)
347 PRINT "LINES";F1-1;"THROUGH";T-1;"HAVE BEEN DELETED":PRINT
350 GOSUB 2040:GOTO 55
399 :
400 REM  (CHARACTER DELETE--ALLOWS YOU TO DELETE CHARACTERS
401 REM  (IN A LINE.  RESPOND "YES" TO DELETE THE CHARACTERS IN
402 REM  (PARENTHESES, <ENTER> TO SEARCH FOR NEXT OCCURRENCE
403 REM  (OF STRING, "NO" TO END SEARCH WITHOUT DELETION)
404 :
405 PRINT"DELETE IN LINE ";:INPUT L:L=L+1:IF L>P THEN 55
415 GET 1,L:VQ$=VF1$:GOSUB 2000:PRINT L-1;":";VQ$
420 PRINT"CHARACTERS ";:INPUT R$
425 B=1
430 N=INSTR(B,VQ$,R$):IF N>0 THEN 435
432 PRINT "'";R$;"' NOT FOUND":PRINT:GOTO 55
435 Q1$=MID$(VQ$,1,N-1):Q2$=MID$(VQ$,N,LEN(R$))
436 PRINT L-1;":";Q1$;"(";Q2$;")";
437 D$="":INPUT D$:D$=MID$(D$,1,1)
440 IF D$="Y" THEN 450
442 IF LEN(D$)<>0 THEN 432
445 B=N+1:GOTO 430
450 Q1$=MID$(VQ$,1,N-1):Q2$=MID$(VQ$,N+LEN(R$),LEN(VQ$)-N-LEN(R$)+1)
451 LSET VF1$=Q1$+Q2$:PUT 1,L:VQ$="":R$="":Q1$="":Q2$=""
453 PRINT L-1;":";VF1$:PRINT
455 GOTO 55
499 :
500 REM  (EXAMINE--USED TO LIST THE CURRENT CONTENT OF A LINE
501 REM  (OR RANGE OF LINES.  USE A LARGE NUMBER (EG: 999) TO
502 REM  (LIST TO THE END OF THE FILE)
503 :
```

Listing 2 continued on page 76

be 29,29. Note that the L command only works on whole lines.

Typing C in response to EDITOR's prompt ACTION? or as the only character following a line number causes EDITOR to branch to the 400-line group. This command is similar to the insert command I in that it operates on part of an existing line. The only acceptable response to EDITOR's DELETE IN LINE? prompt is a line number within the total number of text lines EDITOR is working with. (See listing 6.) Following this, EDITOR will display that line and ask which characters are to be deleted by displaying CHARACTERS?. The user must enter the characters that are to be deleted so that EDITOR may go about searching for their first occurrence. If they are not found, EDITOR simply displays the prompt for a new command.

This infinite loop is useful if the user has inadvertently selected the wrong line. That is, suppose the user enters line number 33 when he intended to select line number 34. EDITOR displays the selected line and asks CHARACTERS?. The user realizes his mistake and simply types some unique group of characters (four-letter words usually work) not contained within the line. When EDITOR does not find these characters in the line, it does nothing to the line before returning to ask for the next command.

If, on the other hand, the typed characters are found, then EDITOR displays the line of text to the search characters, displays the search characters within parentheses, and displays the BASIC input prompt "?". This means that EDITOR is asking if these are characters the user wishes to delete. If they are, the user types

YES and hits return. If they are not, then the user simply hits return and EDITOR starts at that point searching for the next occurrence of the to-be-deleted characters. This process continues until the user agrees to delete the bracketed characters or EDITOR runs out of text line.

Thus, if you wish to delete the third occurrence of the word THE in line 8, you type "8" in response to EDITOR's question DELETE IN LINE? EDITOR would then display the entire text line and ask CHARACTERS? You would then type "THE ", and EDITOR would begin to search the line for the first occurrence of THE followed by a space. Be sure to include the space or EDITOR will obligingly stop at each and every THE (such as in the word THEM) in the line.

When "THE" (including a space) is found, EDITOR displays the line of text to that point, brackets "THE ", and waits. If this is the correct occurrence of "THE ", you should type YES and hit return. EDITOR deletes the article and the space, compresses the text line, and branches back to program line 55 to ask for the next command. (See listing 6.)

If the bracketed word is not the one you want, then hit return and EDITOR will search for the next occurrence of the word. The ability to sequentially search for specific occurrences of a character, of course, is the way you can delete the offending space which EDITOR places in words, should you correct a spelling error with the I and U commands.

Text Manipulation Commands: E and U

There are two commands that allow the user to manipulate the

Listing 2 continued:

```
505 PRINT"EXAMINE LINES FROM,TO ";:INPUT F,T
507 F=F+1:T=T+1:IF F>P THEN 55
510 IF F>T THEN 55
511 IF T<P THEN 515
512 T=P
515 N=0
520 FOR I=F TO T
525 GET 1,I:VQ$=VF1$:GOSUB 2000:PRINT I-1;":";VQ$:VQ$=""
530 N=N+1:IF N<10 THEN 540
535 N=0:R$="":INPUT R$:IF LEN(R$)>0 THEN 60
540  NEXT I
545 PRINT "END":GOTO 55
599 :
600 REM  (UPDATE--THIS COMPRESSES TEXT WITHIN THE SPECIFIED
601 REM  (LINE RANGE TO SLIGHTLY OVER THE LINE LENGTH.  USED
602 REM  (AFTER "INSERT" TO REMOVE PARTIAL LINES.  NOTE--THIS
603 REM  (WON'T WORK TO FORMAT TEXT TO A SHORTER LINE LENGTH)
604 :
605 PRINT"UPDATE LINES FROM,TO";:INPUT F,T
607 F=F+1:T=T+1
610 IF  F>T THEN 55
611 IF F>P THEN 55
612 IF T<P THEN 615
613 T=P
615 N=F
620 GET 1,F:VQ$=VF1$:GOSUB 2000:F$=VQ$
625 QF=LEN(F$):IF QF=0 THEN 629
626 IF MID$(F$,QF,1)="!" THEN F$=F$+" "
627 IF MID$(F$,QF,1)="?" THEN F$=F$+" "
628 IF MID$(F$,QF,1)="." THEN F$=F$+" "
629 F=F+1:IF F>T THEN 685
630 GET 1,F:VQ$=VF1$:GOSUB 2000:S$=VQ$
635 X$=F$+" "+S$+" "
640 IF LEN(X$)>W THEN 650
645 F$=X$:GOSUB 2020:X$=F$:GOTO 625
650 Z=0
655 Z=INSTR(Z+1,X$," "):IF Z>W THEN 665
660 Y=Z-1:GOTO 655
665 LSET VF1$=MID$(X$,1,Y):PUT 1,N:VQ$="":F$="":S$=""
667 PRINT N-1;":";MID$(VF1$,1,Y)
670 N=N+1
675 F$=MID$(X$,Y+2,LEN(X$)-Y-1):GOSUB 2020
680 GOTO 625
685 LSET VF1$=F$:PUT 1,N:VQ$="":S$="":X$=""
686 PRINT N-1;":";F$
690 F=N+1:GOTO 310
799 :
800 REM  (HARDCOPY--USED TO PRINT TEXT FILE (OR PORTION OF IT)
801 REM  (TO PRINTER OR TO SCREEN.  PW=LENGTH OF PRINT LINE;
802 REM  (AFFECTS SPACING WHEN USING CENTERING OPTION.  NOTE:
803 REM  (PROGRAM WILL CRASH IF PRINTER SPECIFIED AND NONE
804 REM  (IS CONNECTED)
805 :
806 PRINT "OUTPUT TO PRINTER (Y) OR SCREEN (N)";:INPUT PR$
807 PR$=MID$(PR$,1,1):IF PR$<>"Y" AND PR$<>"N" THEN 806
808 PW=64
810 PRINT"OUTPUT LINES FROM,TO ";:INPUT F,T:F=F+1:T=T+1
815 IF F>T THEN 55
816 IF F>P THEN 55
817 IF T<P THEN 820
818 T=P
820 PRINT"LINE SPACING (1=SINGLE SPACING) ";:INPUT L
825 IF L*((T-F)+1)>=65 THEN 840
830 PRINT"CENTER TOP-BOTTOM ";:INPUT R$
835 R$=MID$(R$,1,1):IF R$="Y" THEN 845
```

Listing 2 continued on page 77

```
836 IF R$<>"N" THEN 830
840 PRINT"TOP MARGIN ";:INPUT J:GOTO 850
845 J=ABS((65-T)/2)
850 IF W<80 THEN 860
860 PRINT "CENTER LEFT-RIGHT ";:INPUT R$
865 R$=MID$(R$,1,1):IF R$="Y" THEN 880
866 IF R$<>"N" THEN 860
870  PRINT"LEFT MARGIN ";:INPUT K
875 IF K+W<=80 THEN 900
876 PRINT "OVFL":GOTO 870
880 K=(PW-W)/2
900 IF PR$="N" THEN PRINT "                    --- TOP OF PAGE ---"
902 FOR I=1 TO J:IF PR$="Y" THEN LPRINT " ":GOTO 904
903 IF PR$="N" THEN PRINT " "
904 NEXT I
905 FOR I=F TO T
911 IF PR$="Y" THEN LPRINT TAB(K);:GOTO 915
912 IF PR$="N" THEN PRINT TAB(K);
915 GET 1,I:VQ$=VF1$:GOSUB 2000
916 IF PR$="Y" THEN LPRINT VQ$:GOTO 920
917 IF PR$="N" THEN PRINT VQ$
920 IF L<=1 GOTO 925
921 FOR H=1 TO (L-1)
922 IF PR$="Y" THEN LPRINT " ":GOTO 924
923 IF PR4="N" THEN PRINT " "
924 NEXT H
925 NEXT I
930 GOTO 55
999 :
1000 REM  (QUIT--UPDATE NUMBER-OF-LINES RECORD, CLOSE FILE,
1001 REM  (AND END PROGRAM)
1005 LSET VF1$=STR$(P):PUT 1,1
1006 CLOSE 1
1010 END
1995 REM  ---------------- (SUBROUTINES) ---------------------
1997 :
1998 REM  (SUBROUTINE TO TRIM TRAILING BLANKS OFF STRING VQ$)
1999 :
2000 IF LEN(VQ$)=0 THEN RETURN
2005 'IF MID$(VQ$,LEN(VQ$),1)<>" " THEN RETURN
2010 VQ$=MID$(VQ$,1,LEN(VQ$)-1):GOTO 2000
2017 :
2018 REM  (SUBROUTINE TO TRIM TRAILING BLANKS OFF STRING F$)
2019 :
2020 IF LEN(F$)=0 THEN RETURN
2025 IF MID$(F$,LEN(F$),1)<>" " THEN RETURN
2030 F$=MID$(F$,1,LEN(F$)-1):GOTO 2020
2037 :
2038 REM  (SUBROUTINE TO UPDATE #-OF-LINES COUNTER IN RECORD 1)
2039 :
2040 LSET VF1$=STR$(P):PUT 1,1:RETURN
```

text file VF1. The first is the examine command, E. When E is typed in response to the prompt, ACTION?, or as the sole character in a text line (see listings 4 and 6), EDITOR will branch to the 500-line group. At line 505, EDITOR asks which lines of text the user wishes to see by displaying the prompt, EXAMINE LINES FROM, TO?. The user then types in two numbers following the restrictions explained for the delete-line command L, and hits the return key. EDITOR will then display these text lines, until either the video screen is filled or until the last requested text line is displayed.

The number of lines displayed depends upon the capabilities of the particular video terminal and may be altered by changing the IF statement at line 530. Once EDITOR has filled the screen with text, it then displays the BASIC prompt "?" and waits. The user then has the opportunity to review the displayed text.

After this, the user has two options. He may wish to view the remainder of the text originally specified by hitting the return key. Alternatively, the user may decide that he has seen enough, in which case a control command is typed; this causes EDITOR to branch back to line 60, to evaluate the command. When EDITOR has displayed all requested text lines, it will display the word END and request the user's next command.

The other text manipulation command is the update (ie: U) command. (See listing 7.) The purpose of this command is to organize the text material so that each line of text file VF1 contains the maximum number of whole words possible. Remember that the line-length limit was set when EDITOR was started up. This is accomplished by lines in the 600 group. Basically, the procedure *concatenates* (or joins) text lines until the number of characters exceeds that allowed by the line-length limit. To keep things tidy, EDITOR will automatically enter a space between successive lines; if the last character of a line is a period, exclamation point, or question mark, it will enter two spaces.

Basically, the procedure EDITOR uses is one of concatenation of text lines until the number of characters exceeds that allowed by the line length W. Following this, EDITOR checks complete words until the addition of a word exceeds the line-length limit. The beginning of the concatenated line then is the current line less the word that exceeded the line-length limit. EDITOR then uses that word

Listing 3: *Initializing the EDITOR program. When EDITOR is started, it allows the user to work on previously entered text or to type in new text. The length of each text line is also set at this time. (Underlined text represents a user response.)*

```
READY
RUN
EDITOR 03-DEC-79 HT-11 XBASIC 206.04.01
OLD OR NEW TEXT?NEW
LINE LENGTH = 68
CHANGE ?__
LINE LENGTH SET TO 68
ACTION ? A
  1 ?
```

Listing 4: *Using the insert command. EDITOR displays its ability to recognize the location of a set of characters and insert a word at that point. The underlined blank represents hitting the return key.*

```
ACTION ? I
INSERT IN LINE? 8
IN THE LAND OF MORDOR WHERE SHADOWS LIE.
BEFORE ?SHADOWS
IN THE LAND OF MORDOR WHERE ? THE
  9 ?E
EXAMINE LINES FROM, TO ? 8,9
  8 IN THE LAND OF MORDOR WHERE THE
  9 SHADOWS LIE.
END
ACTION ?
```

Listing 5: *Using the delete-lines command. If the range of lines given is valid, EDITOR deletes these lines and moves the following lines to take their place.*

```
ACTION ?L
DELETE LINES FROM, TO ? 3, 4
LINES 3 THRU 4 HAVE BEEN DELETED
```

and all following words of the concatenated line to make the next succeeding line of text. This process continues until the last line specified to be updated has been read and rewritten. Since all of the text has been pushed forward, the last line or few lines specified will be empty. EDITOR deletes these lines and pushes the rest of the text file up to fill the space.

From the user's point of view, the update sequence is started by typing in the single-letter command, U. EDITOR then asks which lines are to be updated with the prompt, UPDATE LINES FROM, TO?. The user must respond with two numbers separated by a comma, as was done for the L and E commands.

Note that the ability of EDITOR to update only certain lines has two advantages. First, it can save update time. Second, and more important, it allows the user to update without disturbing headings, greetings to letters, indented text, or tables within the text material.

The Output Command: H

While EDITOR always saves the text material it is working on when it receives the quit command Q, the only real output command is the hard-copy (ie: H) command. As with the other commands, EDITOR will branch to the appropriate line group (the 800 group in this case) when H is typed.

EDITOR's first job when executing this command is to ready the line printer. (See line 805.) This particular program line is, of course, specific to my system and must be modified for other installations. EDITOR then displays the prompt, OUTPUT LINES FROM, TO?, and the user must respond with two numbers, according to the now familiar restrictions.

Frequently, the entire text file will be printed; however, there are certain advantages to being able to print only a portion of the total text file. For example, some form letters may be designed so that not all recipients will receive all of the information. Or the user may wish to review some portion of the text before continuing to edit the final copy. EDITOR recognizes these desires and allows the user the ability to print only part of the file.

Next, EDITOR asks for the number of blank lines that are to separate individual lines of text. The prompt displayed is LINE SPACING (1 = SINGLE SPACING)?. Here, the user is expected to respond with some number greater than 0. A 1 will single space the text material, a 2 will double space, and so on.

At this point, EDITOR concerns itself with the look of the finished product and checks the length of the text material by computing the total number of lines (text lines plus blank lines). If this is less than a page (65 for my printer), then EDITOR wants to know if the text is to be centered and asks CENTER TOP-BOTTOM?. If the user responds with a YES, EDITOR makes the appropriate computa-

Listing 6: *Using the delete-characters command. EDITOR searches for a set of characters, which, when verified by the user, will be deleted from the line.*

```
ACTION ?C
DELETE IN LINE ?8
  8 IN THE LAND OF MORDOR WHERE THE THE SHADOWS LIE.
CHARACTERS ? THE
IN (THE )?__
IN THE LAND OF MORDOR WHERE (THE )?__
IN THE LAND OF MORDOR WHERE THE (THE )? YES
ACTION ?E
EXAMINE LINES FROM, TO ?8,8
  8 IN THE LAND OF MORDOR WHERE THE SHADOWS LIE.
END
ACTION ?
```

Listing 7: *Using the update command. A separate update command allows editing without the delay associated with a reformat-text routine. This is routinely carried out in more complex text editors after each command. The edited text file is not "cleaned up" until the user is finished with all changes and has time for the operation.*

```
  7 ONE RING TO BRING THEM ALL AND IN THE DARKNESS BIND THEM
  8 IN THE LAND OF MORDOR WHERE THE
  9 SHADOWS LIE.
END
ACTION ?U
UPDATE LINES FROM, TO ?8,9
ACTION ?E
EXAMINE LINES FROM, TO ?7,99
  7 ONE RING TO BRING THEM ALL AND IN THE DARKNESS BIND THEM
  8 IN THE LAND OF MORDOR WHERE THE SHADOWS LIE.
END
ACTION ?
```

tions. If the user responds with a NO, then EDITOR asks TOP MARGIN?; this is a request for how many blank lines are to be between the top of the page and first line of the printed text material.

Finally, EDITOR determines whether the line-length limit W, set by the user, is less than the normal printer line (80 characters on the H14 printer). If it is, then EDITOR wants to know if the text material is to be centered with equal left and right margins and asks, CENTER LEFT-RIGHT?. As with the top-bottom question, a reply of YES from the user causes EDITOR to make the appropriate computation, but this time it is for spaces between the left margin and the beginning of the text line. If the user replies NO, then EDITOR asks for the number of spaces on the left margin before each line of text; the prompt is, LEFT MARGIN?. The user must respond with some number equal to or greater than 0 but less than the number that, when added to the line length limit, will exceed the capacity of the printer. If the printer's line capacity is exceeded, then EDITOR responds with OVFL and again makes the LEFT MARGIN? request. (See listing 8.)

When all of this is established, EDITOR prints the text as requested.

Note that the H14 printer operates on a print buffer that must be filled before printing occurs. If EDITOR runs out of text before the buffer is filled, then it is possible to lose the last line or so of text unless the BASIC program closes the printer file. Line 930 of listing 1 does this procedure for my particular hardware/software combination; it may be inappropriate for other system configurations.

Finally, when printing is finished, EDITOR prompts for the next command and waits. Multiple copies may be obtained by simply typing another H command.

The Quit Command: Q

The final command in EDITOR's repertoire is the quit (ie: Q) command, which may be typed and executed just as any other command. Here the relevant lines are in the 1000 group of listing 1. The Q command allows EDITOR to end in an orderly fashion. To do this, EDITOR first translates the last-line-of-text pointer (ie: P) to a string variable and stores it in VF1(0). This, as previously noted, allows the user to recover the entire text by typing OLD in response to EDITOR's question, OLD OR NEW TEXT?, during start-up. To ensure that the text is properly stored on disk, EDITOR closes the file VF1.

Using EDITOR

Although each individual user will establish his own style of use, a few general comments may prove valuable. First, the update command U takes the longest to execute. The amount of execution time used depends upon the

Listing 8: *Using the hard-copy command. When the text is in its final form, EDITOR gives the user several printing format options.*

```
  5 IN THE LAND OF MORDOR WHERE SHADOWS LIE.
  6 ONE RING TO RULE THEM ALL, ONE RING TO FIND THEM,
  7 ONE RING TO BRING THEM ALL AND IN THE DARKNESS BIND THEM
  8 IN THE LAND OF MORDOR WHERE THE SHADOWS LIE.
END
ACTION ?H
OUTPUT LINES FROM, TO ?1, 99
LINE SPACING ?1
CENTER TOP-BOTTOM ?N
TOP MARGIN ?2
CENTER LEFT-RIGHT ?Y
ACTION ?
```

Glossary

Array: a set of related data which may be accessed as individual elements, by specifying the position of the desired element. Most BASIC systems allow arrays of one dimension (a list) and arrays of two dimensions (a table).

BASIC: Beginners' All-purpose Symbolic Instruction Code. A high-level computer language popular with many users of personal computer systems.

Character: a graphic symbol (as a letter or numeral) used in printing.

Concatenation: (referring to strings) to link or append one string of characters to another.

Character String: a sequence of characters referred to by a single name or variable. See also: String Variable.

Dedicated: set aside or used for one purpose only, as in a dedicated telephone line used only for computer communications.

Disk: a mass storage medium used for computer data, consisting of a flat, circular material with a magnetic coating (similar to magnetic tape) which is recorded on by a specially designed drive unit.

Disk File: a set of related data, as recorded on a magnetic disk.

Hard Copy: graphic images that are recorded on paper in a human-readable form.

Hexadecimal: a number system which uses the base 16 to represent integers. The system uses the numerals 0 thru 9, as well as the letters A, B, C, D, E, and F to represent values 4 bits long in binary.

Integer: any number that is not a fraction and does not include a fraction.

Integer Variable: a computer memory location, having a symbolic name, which may contain only integer values.

Off-line Storage: any mass-storage device which is not presently connected to a computer.

Glossary continued on page 82

amount of text being updated; this may be decreased if the user updates only the minimal number of lines necessary. Also, updates near the end of the text material are accomplished much faster than updates near the beginning of the text material. The reason for this is that there is less text to be manipulated. Thus, while there is some benefit to keeping the update requests to a minimum, there is also a trade-off between minimizing the number of updates and letting an update operation "pile up" behind a large number of otherwise acceptable lines. I have found that updating about every 10 lines seems to be a fair compromise.

Another useful technique involves deleting sentences that are not isolated from other sentences on the same line or lines. One procedure is to first isolate the sentence or sentences using the insert command and then delete complete text lines. As an example, the user first enters the I command to instruct EDITOR to *insert* before the first word of that sentence which is to be removed. In response to EDITOR's BEFORE? question, the user types the first word and hits return. This breaks the text line at the beginning of the sentence. EDITOR then requests another insert, and the user responds with the line number where the to-be-deleted sentence or sentences end and repeats the previous procedure. Following this, the isolated text is removed with delete-lines command, L.

Of course, any number of sequential sentences could be deleted in this manner. An alternative method would involve the delete-characters command, C, to remove the unwanted beginning

Text continued on page 82

onComputing™
GUIDE TO PERSONAL COMPUTING

Name (Please print) _____

Street/Apartment Number _____

City _____ State/Province/Country _____ Postal Code _____

Please start my subscription with: ☐ Summer 1980 ☐ Fall 1980

☐ U.S. 1 yr. (4 issues) @$8.50 ☐ Canada & Mexico 1 yr. (4 issues) @$10.00

FOREIGN (to expedite service, please remit in U.S. funds drawn on a U.S. bank.)

☐ Europe (and all other countries, except above) 1 yr. @$12.00—surface delivery

☐ Bill VISA ☐ Bill Master Charge ☐ Bill me (North America only)

Card No. _____

Expiration _____

Four digits above name — Master Charge only _____

Signature _____ Date _____

Please allow eight weeks for processing. Thank you **4110**

onComputing™
GUIDE TO PERSONAL COMPUTING

Name (Please print) _____

Street/Apartment Number _____

City _____ State/Province/Country _____ Postal Code _____

Please start my subscription with: ☐ Summer 1980 ☐ Fall 1980

☐ U.S. 1 yr. (4 issues) @$8.50 ☐ Canada & Mexico 1 yr. (4 issues) @$10.00

FOREIGN (to expedite service, please remit in U.S. funds drawn on a U.S. bank.)

☐ Europe (and all other countries, except above) 1 yr. @$12.00—surface delivery

☐ Bill VISA ☐ Bill Master Charge ☐ Bill me (North America only)

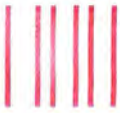
Card No. _____

Expiration _____

Four digits above name — Master Charge only _____

Signature _____ Date _____

Please allow eight weeks for processing. Thank you **4110**



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS Permit Number 45 Martinsville NJ 08836

POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
POB 307
Martinsville NJ 08836
USA

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS Permit Number 45 Martinsville NJ 08836

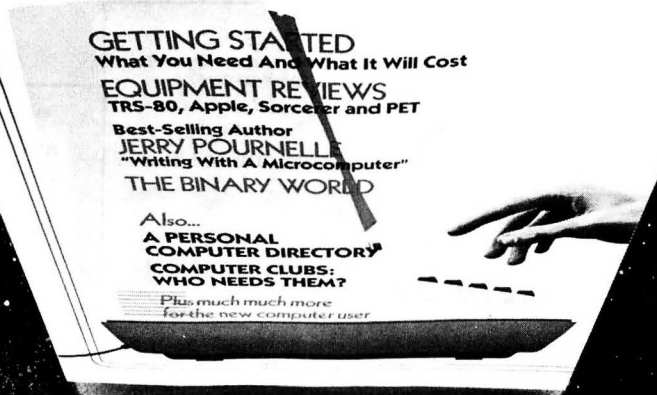
POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
POB 307
Martinsville NJ 08836
USA

A McGRAW-HILL PUBLICATION

onComputing™

GUIDE TO PERSONAL COMPUTING



When will the Personal Computer Explosion touch YOU?

Are you prepared for the explosive transformation? Right in your own home? Electronic mail. Personalized investment analysis. Foreign language tutorial. Home energy management. Robots. Computer music. Secretarial service. Diet and menu planning. And more, more, more.

onComputing™ the new McGraw-Hill quarterly, prepares you for the enormous changes coming during the 1980's (Some are already here). **onComputing™** explains in nontechnical language what personal computers are, how they work, and how you can use them at home, for fun and profit.

Don't let the personal computer explosion catch you off guard. Know what's happening and help make it happen! Prepare now for the exciting future with a subscription to **onComputing™**!

Call Toll-Free
800-258-5485

Start your subscription today.

onComputing™ Subscription Dept. P.O. Box 307, Martinsville, NJ 08836

DOMESTIC subscription rate:
☐ U.S. 1 yr. (4 issues) @ $8.50 ☐ Canada & Mexico, 1 yr. (4 issues) @ $10.00

FOREIGN (to expedite service, please remit in U.S. funds drawn on a U.S. bank.)
☐ Europe (and all other countries, except above), 1 yr. @ $12.00 — surface delivery.

☐ Bill Visa ☐ Bill Master Charge ☐ Bill me (North America only)

Card Number _____ Expiration _____

Signature _____ Name (please print) _____

Street/Apartment Number _____

City _____ State/Province/Country Code _____

Please allow 6-8 weeks for processing.

7OC60

Circle 220 on inquiry card.

Glossary continued from page 80:

Open: (as in to open a file) an initialization procedure necessary to make storage space available on a magnetic disk.

Prompt: a special character displayed by some computer programs to show that they are awaiting operator input.

Return: a key on the standard ASCII keyboard used to indicate that all instructions on one line have been typed. This is used as an indicator to most computers that typing of the desired command has been completed.

String Variable: a special computer-memory location that is intended to hold only a sequence of characters as a value. It is referred to through a symbolic name.

Virtual Files: files which appear to reside in computer memory, but may actually be recorded on magnetic disk (or other mass storage device). The operation and maintenance of virtual files is usually performed as a housekeeping chore by the computer and is transparent to the user.

Word Processor: a combination of programs that allow the operator to enter, manipulate, add, change, and delete text before it is printed in final form.

Text continued from page 80:

and ending of a sentence or sentences. Unfortunately, this method often involves typing in a considerable number of characters which, if not typed correctly, will not be deleted. I have used both methods and found the former to be faster and more accurate.

Finally, it may be mentioned that the modular organization of EDITOR allows individuals to customize EDITOR to their own needs. Some users may want to specify the name of the text file instead of letting EDITOR do it at program line 10; then it is possible to use EDITOR to access multiple source files. A modified EDITOR including appropriate tab functions can produce a text file that is in the appropriate format for some computer languages—FORTRAN, for example. However, when this is done, you must be careful when using the U command, since EDITOR will be most happy to combine one or more instructions on a single line, if they are within the line-length limit. Also, the insert command, I, is somewhat of a problem since it breaks lines (or instructions, in this case). For the most part, the user is required to limit himself with line editing (except for the delete-character command, C). But, then this is what most BASIC interpreters do anyway.

I hope that you will find some use for this text editor. Granted, it can be slow, but it costs much less than commercial text editors. In any case, the structure of this program, when studied, can give you an idea of how text editors work.■

About the Author

T. E. Levere is a professor in the Psychology Department at North Carolina State University.

A Writer Looks at Word Processors

by Jerry Pournelle

Jerry Pournelle (who tells about "Living With Computers" in this issue) gives us a no-holds-barred evaluation of several word processors he has used.

Since I write for a living, when I decided to move in with a microcomputer I naturally worried about word processing before I looked at the other attractive features of computers. As it turns out, that would have been the proper approach even if I sold widgets for a living.

Anyone in business is a professional writer.

True, most don't know it; yet it remains true. Think about it. What do you do with your time? I don't care what your business is—if it's at all complex you'll spend a decent percentage of your time putting words on paper. A very great deal of business activity consists of answering correspondence, ordering supplies, negotiating prices, keeping notes and memoranda, generating contracts, and generally dealing with business papers. Large businesses can hire people to do a lot of that; the small businessman is going to have to do it himself, and to the extent that he avoids "paper work" he is neglecting his business and losing money.

Those unfamiliar with a good word-processing program simply cannot understand how much easier paperwork becomes when you have one. Thus I can say quite pontifically: if you have any idea of using your computer for a small business, be certain that it can handle the very best word processors. Darned near any decent computer will run BASIC and FORTRAN and a number of data-base languages and fair-to-decent accounting systems. They'll all keep track of inventory and do at least rudimentary file control (and the most elementary computer file system beats the devil out of everything but a very experienced secretary). They'll all play games, and a lot of them will give you pretty graphics.

Unfortunately, not all of them will do good word processing, and some are so crippled they'll *never* be able to accept a good text editor.

If it can't run a really good text editor, you're losing half the value of the computer.

I mean that quite seriously. Certainly I'm prejudiced in favor of machines that handle words gracefully; I make a living out of words. But I'm not now speaking from prejudice but experience.

Do understand that I'm talking about a computer, not a "word-processing machine," which in my judgment is the abomination of desolation. I have never seen a "dedicated word processor" that could handle words better than Ezekiel (my Cromemco computer system) can. (I am talking about a special-purpose machine here; there are "word-processing programs" for real computers.) The best equal what he can do, and so what? "Word Processors" can't crunch numbers or keep files or deal with the taxes or (I do have some special problems) design planetary systems. "Word Processors" aren't worth the gasoline it takes to haul them out to the city dump. Don't even contemplate buying one. Get a real computer, which won't cost much more than the "Word Processor," and will do a lot more besides.

But do be certain that your real computer can mash words properly, lest you end up working too hard like my science fiction writer friend Gordy Dickson.

Words, Words, Words

What do I mean by the proper mashing of words?

First, let's make something perfectly clear: any computer system worth buying at all will be impressive compared to a typewriter. The worst of the text editors—and there are some very bad ones indeed—will generally outperform most typing machines, and thus tempt inexperienced buyers. If your entire experience has been with a Model T Ford (ie: a typewriter in this rather fanciful analogy), then a Model A will seem very attractive, and you'll love it until you find that for not a lot more you could have bought an Alfa-Romeo; and in fact the best text editors are to the usual ones as the Alfa is to the Model A.

At this point I'm frustrated. I'm a professional wordsmith, and words are failing me. I could *show* you the differences between various text editing systems in five minutes; but I'm not sure I can describe them. However, I'll try.

The lowest level of text editor —by text editor I mean a *program*, and remember that not all programs will run on all computers—goes by the unlikely name of TECO.

TECO editors are better than typewriters, but not much. They're line-oriented; you have to tell them everything you want to do. For example, if you want to kill three lines of text, you type the command 3K, and the machine dutifully deletes the next three lines. (It's hard cheese if one of those was a line you wanted to keep.) Having done that the machine just sits there. It doesn't show you what you've done. To get a display of the text with the dead lines gone you have to give another command, such as 5T meaning "type" the next five lines.

There are other problems. TECO editors don't tell you where the cursor is, which is to say that unless you understand what the editor is doing it won't tell you where in the text the next word is going if you want to insert something.

In other words, they're horrible.

So why have they survived? Well, first, for historical reasons; TECO editors were among the very first to be programmed. Second, there are a lot of them around, and often they're given away; as an example, if you buy the CP/M operating system from Digital Research Corporation, they will give you a TECO. (They charge about what it's worth in my opinion.) Third, programmers have become accustomed to them. TECO editors have interesting features; in the best of them you can actually write mini-programs which the machine will execute, and that can be quite useful to people who are accustomed to them.

What they can't do is to painlessly create text because they're not really invisible. You have to think about what you're doing, and if you think about the editor you aren't concentrating on what you're writing, and that's not only death to creative effort, but also a lot of work.

People who have never seen any text editor other than a TECO can't understand why anyone gets very excited about writing with computers, and no wonder.

At the top of the tree we have the full visual editors such as Electric Pencil. With the Electric Pencil you don't think about what you're writing; you simply write. As you type, the words appear on the glass screen, and when you get to the end of a line you don't bother with a carriage return because the computer keeps track of all that for you; if the last word on a line is too long to fit, the machine pulls the word down to the beginning of the next line. To insert text you give a simple command and begin typing where you want the inserted stuff to go; everything after that marches across and down the page to make room. Alternately, you may type over words to change or correct them.

Until recently, Electric Pencil (Michael Shrayer Software, costing about $300) had no real competition even though it has known bugs and its owners are unresponsive to user suggestions and complaints. Despite all of its problems, for years it was far and away the best editor around if what you wanted to do is sit down and write. Pencil can be learned in a matter of hours, learned well in days, and becomes "natural" within a couple of weeks; and when it reaches that stage the results are astonishingly good: the editor vanishes, so that you can ignore *how* the words get recorded and concentrate on *what* you want to say. Pencil really can take all of the mechanical work out of writing.

But now Electric Pencil does have competition: I've recently received a review copy of Magic Wand from Small Business Applications Inc of Houston, Texas (available only through dealers, costing about $400), and it cures many of Pencil's more vexing

features. Another company, Software Services (18323 Vanowen St Suite 34, Reseda CA 91355), produces an excellent but relatively expensive fix called Pencil Sharpener that makes Pencil much more useful to the business firm that handles a lot of mail. (Magic Wand does approximately what Pencil plus the Sharpener will do.) Unfortunately, although for many applications Magic Wand is preferable to Pencil, there remain features in Pencil that Magic Wand doesn't have. I'm particularly unhappy with Wand's disk operations, which are reliable but almost unbearably inconvenient. Sigh. I'm using both Pencil and Wand to write this article, and I honestly can't choose between them; each has both advantages and problems, and both lack essentials. There is one major difference: SBA Inc is far more responsive to user suggestions for fixing up their product than Pencil's author.

Then there's the Proteus Engineering word processor which my friend Tony Pietsch, the computer genius who put my system together, says he'll have running "Real Soon Now." It's supposed to do *everything*, and it just might; certainly Tony's other software works splendidly. But it's not done yet.

After Pencil and Wand there's everything else.

The Electric Pencil is spectacularly useless as a programming editor. Magic Wand will do, but really, for programming applications there's only one real choice, MicroPro's WordMaster, at a discounted price of $125 easily the best software bargain around. In a pinch WordMaster could be used to create text; Gordon Dickson has been using it for several months now because his engineers built him a system that can't run Pencil. For all of Word Master's faults, he'll never go back to a typewriter. However, if you're creating text with WordMaster you're working about twice as hard as you should.

I've tried about a dozen programmer-oriented editors, and for programming WordMaster is in a class by itself; really superb.

I've tried about a dozen programmer-oriented editors, and for programming WordMaster is in a class by itself; really superb. Having created WordMaster, MicroPro tried to improve it. The result was their very expensive WordStar (costing $445 discounted). Until I saw Magic Wand, if I were allowed to own one and only one editor, Word Star would have been it; but it's no more than second best for either programming or text creation, and it costs more than Pencil and WordMaster combined.

WordStar has a lot of features; as the British say, too many by half if you ask me. I am told by people I respect that after you get used to it, WordStar is very nice indeed, and I suppose I believe it; but I also know it takes a *lot* of getting used to. You might not

want to pay for the time it would take to learn it. Worse, from my limited experience with it, is that WordStar got in the way of creativity. I kept stopping to think how to use the editor instead of thinking of what I wanted to say. About half of the commands take a second or so plus two or more keystrokes to implement, again interfering with thought processes. WordStar doesn't automatically format after you insert and delete lines or words. I would recommend WordStar only to someone who can try it before buying it. Some will like it a lot, others will find it distracting. My personal preference is for Pencil or Magic Wand for text creation. If you're doing much programming (and you'll be surprised, you will), get WordMaster as well.

There's one other application of WordStar: if you're publishing newsletters and need exact formatting so that you can insert diagrams and illustrations, nothing I've seen comes close; WordStar lets you see on the screen the precise format that will be printed on paper, and you don't have to insert all kinds of diacritical remarks into the text to get the layout you want.

Electric Pencil, for all its problems, is very convenient in printing features; you can get a hard copy of your text at any time from right in the program, rather than having to put in formatting characters and then invoke a separate print routine.

So What Does It All Mean?

If you accept my thesis that the ability to properly mash words is the key criterion in choosing a computer, and further accept my fairly wide experience that there are no word-mashing systems to rival WordMaster, WordStar, Magic Wand, and Electric Pencil, then you're a long way towards choosing your computer.

To explain why, we have to get technical.

There are two basically different and totally incompatible ways for computers to talk to you: terminal and direct memory access (DMA) systems. (Actually there is another; you *can* get a system that doesn't put words on a glass screen, but instead types them on paper. Such systems can be fun and they'll play a lot of games, but they have such severe limits that they are mostly of interest to hobbyists—broke hobbyists.)

Of the two ways to get words up on a screen, the most common is through use of a terminal: one of those all-in-one gadgets that combines screen, video display board, and keyboard. Popular makes are the Lear-Siegler ADM series, and the various Hazeltines. A terminal has its own memory quite separate from the main memory of the computer. Words typed on the terminal keyboard go into the *terminal's* memory, and are duly echoed onto the terminal screen. They are then sent across to the computer through a port. The computer does things to the text, then rewrites the result onto the terminal screen. This can be slow or fast; if slow it's usually intolerable, and even when done quickly you are almost always *aware* of a delay between what you do and what you see.

A direct memory access (DMA) system works quite differently. In a DMA system there is a board inside the main computer box. This board takes a chunk of the computer's normal memory and displays it; it's like a small window into memory. Thus what the computer knows is what you see, and that happens instantaneously.

The chief advantage to terminals is that they're fairly easy to connect to the system. They're standard and are often installable by relatively naive users. Also, they don't use system memory to store displays.

One disadvantage is that they're not really "transparent"; you see them operate, and that can get in the way of text creation. Another disadvantage is that with a terminal system you can't use Electric Pencil. You can use Magic Wand, WordStar, or Word Master, but they'll be slower on a terminal than on DMA.

In my judgment, terminals make text editing from somewhat to very inconvenient, and that's just too great a disadvantage to put up with.

So: shop for a system that will allow you to install DMA, and accept no substitutes. Don't let them tell you that a very high-speed terminal is "just as good"; in my judgment it simply isn't. Even at the very highest speeds, you will see your screen flicker annoyingly as it gets rewritten every time you type in some text; and believe me, a couple of hours worth of that is far more than enough.

At present the only text editor which makes use of all of the nifty features of DMA is Electric Pencil (which runs on DMA systems *only*). The others—Magic Wand, WordMaster, WordStar — *will* run on DMA, but they scroll rather badly. The Real-Soon-Now Proteus editor will, at least in the first rendition,

specialize in DMA and use all of the DMA features.

If you want the best in text editors, there's another limitation.

The best editors run only on 8080 or Z80 systems. They haven't been implemented for the 6800+ series computers, and the best editors available for any 6502 series computer are comparable to WordMaster.

This means that the Apple, for all its popularity and all of its excellent features, can't really compete. [*Microsoft's new Z80 Softcard for the Apple II allows CP/M programs to be run on the Apple...***ed]**

On the other hand, the TRS-80 will run Electric Pencil. It doesn't do it perfectly, and the TRS-80 has other limits, but you can get Pencil running on one. In order to do so, you'll have to have your TRS-80 modified to do lowercase characters and have a control key installed; this will cost about $75 and will void your warranty. I have also seen Tandy advertisements for a new TRS editor and lowercase modification, but I can't comment on its suitability until I've tried it.

There is one fix for the TRS-80 Model I (Level II) that will let it run the CP/M operating system on 8-inch disks. It's a set of boards from Omikron Systems (1127 Hearst St, Berkeley CA 94702) that truly turned the TRS-80 (my auxiliary computer) into a 48 K "normal" memory-mapped machine that will run all of the important text editors. My secretary uses it with the Omikron fix daily, and it works fine. Even without the Omikron additions, the TRS-80 is a DMA computer and thus not inherently prevented from running top-of-the-line editors.
rather than one with DMA (and terminal-type systems are often more convenient to buy), then please, please make certain you see it in operation with the text editor you intend using. If you don't, you may be in for a lot of inconvenience; until you've used a good text editor, you'll never know what being happily married to a computer can be like.

Is This Guy Serious?

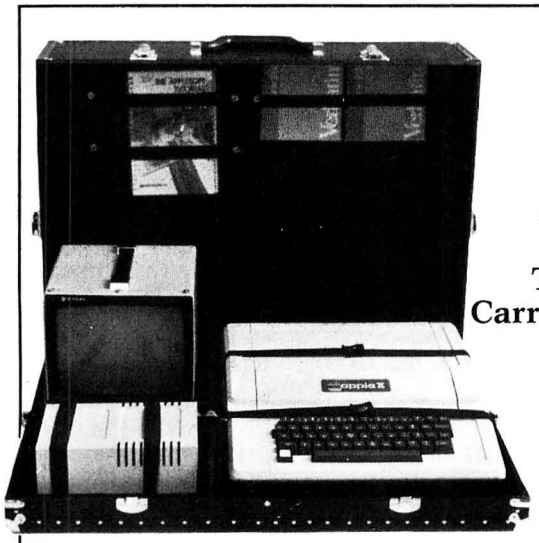
Do I really mean that if a computer can't do painless text editing the machine is not worth having?

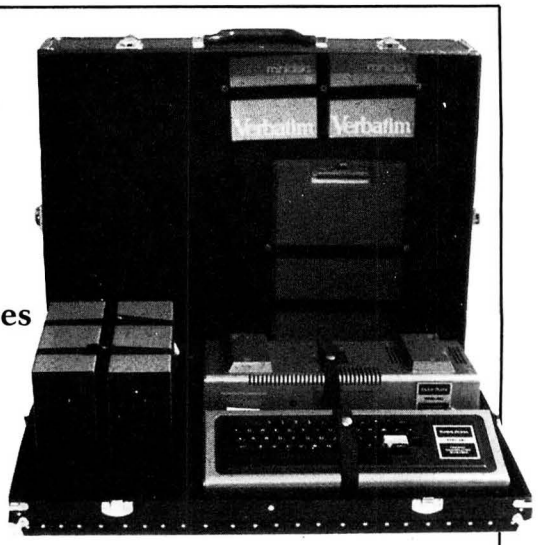
That may be overstating the case a bit, but I guess I do mean that. Certainly I mean it as serious advice to professional writers. If you accept anything less in the way of a text editor you will almost certainly regret it. And I guess I mean it for anyone contemplating using his or her machine in business, because in my experience no matter what the business you'll spend a *lot* of time mashing words, and why settle for less than the best? In my opinion, there's nothing other comparably priced systems can do that a good Z80 with DMA can't accomplish, and nothing else can do editing quite as well as a DMA Z80 can.■

newProducts

Conducted by Charles Freiberg

These pages are designed to keep our readers in touch with the marketplace. The material which appears here is obtained from manufacturers and is not to be taken as an endorsement by onComputing. We invite manufacturers to submit material and we publish the information we feel will be of interest to our readers.



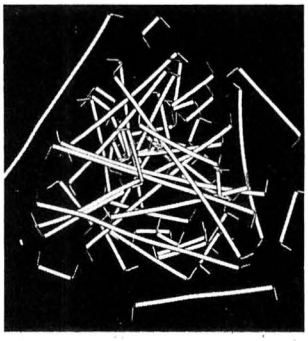


Apple and TRS-80 Carrying Cases

Custom-designed carrying cases for the Apple and TRS-80 systems are now available. The TRS-80 case contains room for the TRS-80 keyboard, expansion interface, two disk drives, power strip, two boxes of floppy disks, and manuals. The Apple case can hold the Apple computer, the 9-inch Sanyo monitor, two disk drives, power strip, two boxes of floppy disks, and manuals. The cases are finished in black vinyl with metal reinforced corners. The interior is lined with protective foam rubber covered with black velveteen. The cases are configured so the system may be operated in the case. Each case weighs approximately 17 pounds and costs $179. Contact Computer Textile Inc, 10960 Wilshire Blvd, Suite 1504, Los Angeles CA 90024.

Circle 270 on inquiry card.

Jumper Wires for Designers



This kit contains 20 AWG (0.8 mm) jumper wires that are precut and bent ninety degrees at both ends for easy insertion in breadboarding modules. The jumpers are Kynar insulated and prestripped 6.35 mm (0.250 inch) on each end. These jumpers are offered fifty per package in lengths from 13 mm (0.5 inch) to 100 mm (4 inches). The RW-50 kit is priced at $2.95 from local electronics dealers or directly from O K Machine and Tool Corp, 3455 Conner St, Bronx NY 10475, (212) 994-6600.

Circle 271 on inquiry card.

Catalog for Micah Software

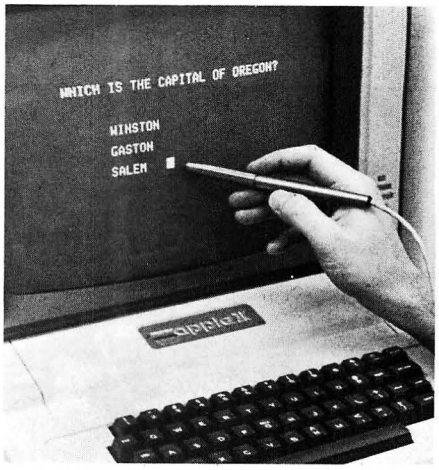
A four-page foldout catalog lists software from Micah, POB 22212, San Francisco CA 94122. Some of its software products include Expand, which expands CP/M to run Cromemco software; Spool, a spooler for CP/M or Cromemco; CBIOS, CP/M for Cromemco computers; and DUP/1, disk utilities for CP/M and CDOS. They also have Osborne business software and graphics software. Contact the company for a copy of the catalog.

Circle 272 on inquiry card.

Sink the Bismarck

Computer Bismarck is an historical simulation game of the British attempt to find and sink the German battleship Bismarck in 1941. The game is played on an Apple II with Applesoft read-only memory (ROM) or an Apple II Plus and it requires 48 K bytes of programmable memory and a floppy-disk drive. It features high-resolution color graphics and can be played by one or two players. Players take turns moving their vessels and aircraft across the North Atlantic. Only enemy units which are spotted are revealed to the players. Rules cover all of the critical aspects of the naval campaign, from weather to ship fuel capacities. Combat occurs when opposing units have spotted each other. Computer Bismarck comes with a program disk, rule book, and seven player aid charts for $59.95, from Strategic Simulations Inc, POB 5161, Stanford CA 94305.

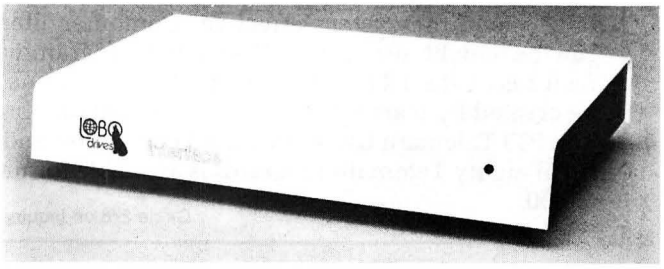
Circle 273 on inquiry card.



Light Pen for Apple II Users

The Lipson Light Pen is now available for the Apple II. The pen is packaged with twelve BASIC programs on cassette, a manual, cable, and a connector to PDL(0) on the Apple II. The demonstration programs are designed to be incorporated into programs designed by the user. The pen utilizes a cadmium selenide cell for light detection, enabling the user to detect and measure varying intensities of light. High-resolution graphics, sound, and color are implemented in the demonstration programs. The Lipson Light Pen is available exclusively from ARESCO, POB 1142, Columbia MD 21044, for $24.95.

Circle 274 on inquiry card.



Lobo Drives Offers Expansion Interface for TRS-80

Lobo Drives International, 935 Camino Del Sur, Goleta CA 93017, announced the addition of an enhanced expansion interface for the Radio Shack TRS-80 personal computer.

The Model LX80 can expand memory storage capacity up to 40 megabytes. It provides facilities for up to 32 K bytes of programmable memory and offers a second serial port. The keyboard read-only memory (ROM) can be overridden for booting in diagnostics and customized operating systems. There is a bidirectional parallel port exclusively for the Lobo Drives Model 7710T Winchester hard-disk drive. Other features include a parallel Centronics printer port; screen printer port; two microprocessor-controlled bidirectional serial ports; and a crystal-controlled real-time clock. The Model LX80 Expansion Interface is priced at $525.

Circle 275 on inquiry card.

newProducts

Educational Computer System

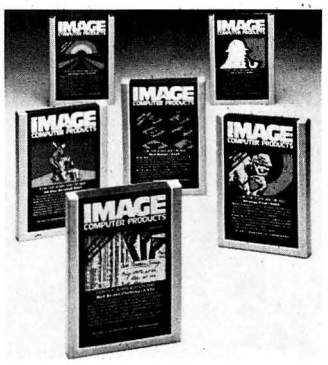
The Validated Computer Math System from Psychotechnics Inc, 1900 Pickwick Ave, Glenview IL 60025, features eighty Telemath programs which are cross-referenced to three of the leading math basals. The PTI Telemath System can be used as a supplementary program for kindergarten thru the eighth grade and as a remedial math program at the secondary or adult level. Each Telemath unit can support from one to five students at a time. The graphics features and the game-type structure of the programs have made the system effective. Computer literacy can be taught using BASIC and the programmable features of the PTI computer. Additional lessons can be created by teachers to add to the Telemath library. The PTI Telemath Lab with one PTI computer and the set of eighty Telemath programs is available for under $3500.

Circle 276 on inquiry card.

Quay Computers

The Quay 500 and 520 systems utilize the Z80 microprocessor. They use Quay's 94F/MPS single-board computer and include 32 K bytes of dynamic programmable memory (expandable to 64 K bytes); two 5-inch floppy-disk drives; the CP/M 2.0 operating system; an RS-232C or 20 mA serial port, and a Centronics-compatible line-printer port. The Quay 500 system provides a formatted disk capacity in excess of 400 K bytes and costs $2500; the 520 system has a disk capacity of more than 800 K bytes and is priced at $3200. Contact Quay Corp, POB 386, Freehold NJ 07728, (201) 681-8700.

Circle 277 on inquiry card.



Atari and Texas Instruments Software

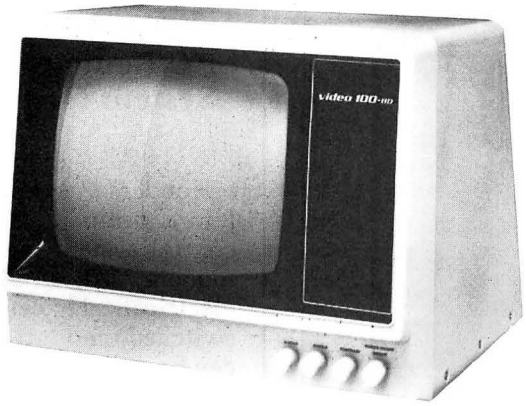
Image Computer Products Inc, 615 Academy Dr, Northbrook IL 60062, has introduced a series of programs for the Atari 400 and 800 series and the Texas Instruments 99/4 microcomputers. The programs include Baseball, Wall Street Challenge, Mind Master, Strategy Pack, Skill Builder, and Tournament Brick Bat. There are two copies of each program, which arrive on cassette. Some of the simulation games allow users to save the program on tape in the middle of a game so they can resume playing later. The prices for the programs are $19.95 and $29.95.

Circle 278 on inquiry card.

TEA: An 8080/8085 Co-Resident Editor/Assembler

This book describes the use of Tychon's Editor/Assembler (TEA) to generate assembly-language programs for 8080 and 8085 microprocessors. TEA is used to create, modify, and assemble source programs. Written by Christopher A Titus and published by Howard W Sams and Co, 4300 W 62nd St, Indianapolis IN 46268, the 256-page, softbound book is $8.95.

Circle 279 on inquiry card.



Twelve-Inch Monitor for Under $200

Leedex Corp, 2300 E Higgins Rd, Elk Grove Village IL 60007, has introduced a 12-inch black-and-white monitor, the Video 100-80. Built for industrial use, the monitor includes a metal cabinet and a removable face plate that provides mounting space for a floppy-disk drive. There is also space inside the cabinet for an 11 by 14 printed-circuit board for custom-designed electronics. The 90-degree deflection picture tube allows an 80-character by 24-line display, and the unit features a 12 MHz band width.

The Video 100-80 is plug compatible with Apple, Atari, Radio Shack, OSI, Microterm, and Exidy computers. It is priced under $200.

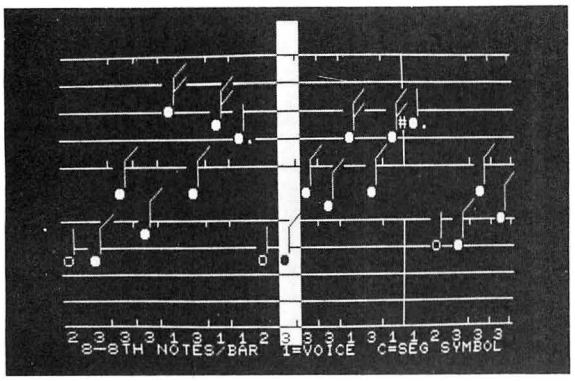
Circle 280 on inquiry card.

Eight-Inch Floppy-Disk Controller

Disk 2+2 is a single-density, 8-inch floppy-disk controller for the Apple II computer. It increases the data on line, increases the individual file size, and reduces the number of disks handled by the user. The board operates under the Apple disk operating system 3.1 or 3.2. It will control up to four standard, 8-inch floppy-disk drives. The card uses a 1771 LSI controller integrated circuit that allows exchange from the Apple to IBM 3740 format. Disk 2+2 costs $400 and is available from Apple dealers. For more information, contact Sorrento Valley Associates, 11722 Sorrento Valley Rd, San Diego CA 92121.

Circle 281 on inquiry card.

newProducts



Four-Part Music System for PET

A B Computers, 115 E Stump Rd, Montgomeryville PA 18936, has announced a system enabling PET users to create and play musical compositions of up to four parts. The KL-4M board includes an 8-bit digital to analog (D/A) converter, a low-pass filter, and an audio amplifier. No additional hardware other than a speaker is required. Connection is made via the PET parallel and cassette ports. The KL-4M is compatible with any of the four-part music monitors. The Visible Music Monitor is written in 6502 machine language and displays the musical staff and notes for all four voices on the PET screen. The KL-4M includes edit capabilities, successive song loading without intervention, user definable keyboard, tempo flexibility, transpose capability, and waveform modification capability. Music can be played with or without note display. The entire system is $59.90. The KL-4M board is $34.90 and the Visible Music Monitor is $29.90.

Circle 282 on inquiry card.

Sound Generator for the Apple II

Symtec Incorporated has introduced a sound synthesizer card for the Apple II. The Super Sound Generator, or SSG, uses thirteen programmable registers to control three voices. The SSG is provided with an output cable and RCA phone plug for hook-up to stereo systems. It features separate 8-bit parallel input and output ports for connection to accessories, such as an alphanumeric keyboard, an organ keyboard interface, or a parallel printer driver. SSG control is accomplished with a series of four POKES to three memory locations. The SSG card may be programmed in any language available for the Apple. The music composing software provides for entry and editing of the entire music score using keyboard commands. The score can be copied by a graphics printer. Stereo effects and orchestrations can be produced using multiple cards. Up to twenty-one voices can be accessed by the user with a complete complement of SSG cards. The Symtec SSG is available for $159.95 from Symtec Inc, POB 462, Farmington MI 48024.

Circle 283 on inquiry card.

AC Remote Control System for the PET

Honders Incorporated offers a complete AC remote-control system for the Commodore PET or CBM. Most electrical devices can be switched on or off and lamps can be dimmed or brightened under computer control. No additional wiring is needed. Up to 256 points can be controlled. This system is useful for security and energy control systems. The basic package includes a plug-in module to the PET's second port, three remote power controllers, and a cassette software package for demonstration and applications. The package may be ordered for $179 from Honders Inc, Kennel Rd, Cuddebackville NY 12729.

Circle 284 on inquiry card.

Software for Music Board on CP/M Compatible Disk

Software support for the Newtech Model 6 music board is available on CP/M compatible disks. The MV80 Multivoice Music Interpreter allows the user to enter four-voice music in a simple notation. The waveforms for each voice can be individually controlled to create the impression of a quartet composed of different instruments. MV80 requires CBASIC2 and a 40 K-byte or larger 8080, Z80, or 8085 CP/M system. MV80 is available on 8-inch floppy disks for $29.95 including a manual. Contact Newtech Computer Systems Inc, 230 Clinton St, Brooklyn NY 11201.

Circle 285 on inquiry card.

Line Printer Software for the TRS-80

Line Printer upgrades any BASIC program that generates printed reports. It requires 32 K bytes of programmable memory, a floppy-disk drive, and a Centronics printer. The main feature of the program is that it can set aside up to 16 K bytes of memory as a buffer, enabling the computer to send output to the printer while continuing to work on the program. Paging operations are provided. Line Printer can print the screen when the user types shift/up-arrow. This program will list programs with single statements on a line. The program is resident in memory during the running of BASIC programs and can be accessed at any time. The price is $24.50 and is available from The Bottom Shelf, POB 49104, Atlanta GA 30359, (404) 491-7567.

Circle 286 on inquiry card.

Computer Business Forms

Speediform is a multipurpose, multipart continuous form set. The first and last parts of this set are pasted together in the extended margins on both sides; additional margin bonding allows extraction of the complete form set intact. The new product can be used on typewriters, serial-character printers, and high-speed printers, and accommodates any after-writing procedure. Speediform incorporates a smudge-free carbonless paper, which allows subsequent entries to be made any time after printout. Order Speediform from Moore Business Forms Inc, 1205 N Milwaukee Ave, Glenview IL 60025, (312) 291-8227.

Circle 287 on inquiry card.

Word Processing for the UCSD Pascal Operating System

Renaissance Systems Inc, 11760 Sorrento Valley Rd, Suite M, San Diego CA 92121, has announced two software packages for the UCSD Pascal operating system. PROFF is a program to format and print text files. The package features adjustable margins; filling, centering, and adjusting; automatic page numbering; text underscoring and printing. An "include" command allows reading from files other than the original input file. FORML aids the user in document generation. Multiple copies of a form letter, each addressing a different person, can be produced. FORML requires a PROFF source file to perform textual substitution and then calls PROFF to produce the modified copy of the document. The packages currently support output to a Diablo Hytype II printer, a disk file, the system console, or the system printer. The packages are available in machine-readable form on an 8-inch, soft-sectored, single- or double-density floppy disk. Manuals are included with the package and can be purchased separately for $25. PROFF costs $425 and the package of PROFF and FORML is $500.

Circle 288 on inquiry card.

Letters continued from page 7:
with Apple and Wang computer systems (much of which was self-taught), and after my incoherent stumblings through these systems I now wish your magazine had been on the spot with me then. Although I now possess a better-than-average understanding of computers, I can understand how one with little or no knowledge of these "brain amplifiers" could step right into a system with minimal effort using your magazine as a ready reference.

Of particular note are the contributions of Elizabeth M Hughes. Were I to start all over again, I would certainly get my hands on anything she has promulgated. Her efforts toward making seemingly mountainous information understandable to any novice are head and shoulders above anything I have previously witnessed. The only avenue you have not pursued is comparison studies of various systems, components, etc. For those not yet in the field and for those who have yet to purchase their own personal computer, I would imagine that this facet would be greatly appreciated.

All in all, you have cost me eight to ten hours sleep over the past two nights (I just can't seem to put the darn thing down). Congratulations on an excellent publication and I look forward to your continued fine service.

Stephen F Swanson
Okinawa JAPAN

Business Computer Update

Dear Editor:

In regard to the article appearing in your Spring 1980 issue by Charles Freiberg entitled "A Small Business Computer Directory," the listing for AlphaMicro Systems appears to be in error. As AlphaMicro dealers for the last three years, we have found the AlphaMicro Systems AM-1030-01 to be the most popular system in their product line. The specifications as supplied in AlphaMicro Systems price list, dated November 28, 1979, are as follows:

Model: AM-1030-01
Price: $16,175
Processor: WD-16 (16-bit processor)
Memory: 64 K bytes, expandable to 256 K bytes
Standard Equipment: 10-megabyte cartidge disk, 64 K dynamic programmable memory, six RS-232 I/O ports, AMOS operating system (multi-user), 16-bit bus structure, cabinet.
Additional Hardware and Software: Alpha Accounting Package, including Accounts Receivable, Accounts Payable, General Ledger, Payroll, Order Entry, Word Processing, etc.

Russell T Grele
JRV Corporation
Hamden CT

Dear Editor:

Mr Freiberg's compilation of "A Small Business Computer Directory" in your Spring 1980 issue certainly provides a valuable service to both present and prospective owners of small computers. May I add to the list by calling your attention to a few omissions.

Hardware:

Intertec Data Systems Superbrain.
Radio Shack TRS-80 Model II
Heath Company's H89 does not include two I/O ports and printer interface as standard equipment. They are options. Also, Heath offers an MBASIC on 5-inch disk.
Zenith Data Systems Z89 (equivalent to Heath WH89) now comes only with 48 K of programmable memory and at a higher price, of course.

Software:

Aardvark Software offers quite a list of tax software.
Clark Systems Corporation, POB 490156, Atlanta GA 30349, offers General Ledger, General Ledger II, Mailist, and Mailpro, all for Heath or Zenith H89.
Microsoft offers COBOL-80.
Micropro International was listed, but no mention was made of the many programs they offer, such as Super-Sorts I, II, and III, WordStar, WordMaster, etc—all available from Lifeboat Associates.
Phase One Systems, 700 Edgewater Dr, Suite 830, Oakland CA 94621, offers the OASIS line of software for an extensive list of hardware systems. Since these are operating system enhancements, perhaps they don't qualify as business computer software, but the efficiency and convenience of operating systems is not to be ignored without cost.

Lloyd E Johnson
Leejon Engineering Company
East Peoria IL

Error Message

We cheerfully wish to inform our readers of a pricing error in the Spring 1980 issue. The price for the *Guide to TRS-80 Information* is *not* $3 as printed, but $2.80, plus $0.50 postage and handling. A discount price for quantities is also available. Contact F E Huebner, POB 37206, Oak Park MI 48237.

Coming Events

June-August, Introduction to Microcomputers, Oklahoma State University, Stillwater OK. Approximately two-thirds of this workshop is devoted to hands-on instruction using the PET 2001 with one microcomputer station per two participants. Contact Technology Extension, Oklahoma State University, Stillwater OK 74078, (405) 624-5714, for schedules of the two-day workshops.

July 7-11, Computers and Related Products, Hyatt Regency Hotel, Seoul SOUTH KOREA. This show is limited to approximately forty firms for exhibition. For details, contact Robert Wallace, Rm 6015A, U S Dept of Commerce, Industry and Trade Commission, Washington DC 20230.

July 13-19, Educational Applications of Microcomputers, Mt Herman Campus, Northfield Mt Herman School, Northfield MA. This course is part of an ongoing program to instruct teachers, school and public librarians, and media specialists in the use of microcomputers for educational use. Arrangements are being made for graduate-level credit through Boston State College. Contact Stacey E Bressler, Field Coordinator, Massachusetts Educational Television, 54 Rindge Ave Extension, Cambridge MA 02140, (617) 876-9800.

July 22-24, Microcomputer Show, Wembley Center, London ENGLAND. New products will be exhibited, along with presentations of papers. For information, contact TMAC, 680 Beach St, Suite 428, San Francisco CA 94109.

August 4-6, Data Entry Management and Supervision Seminar, Chicago IL. Data entry managers and supervisors will benefit from the techniques provided in this seminar. Topics will range from data entry control techniques and improving data entry operator productivity to personnel communications and motivation. Contact MIC, 140 Barclay Ctr, Cherry Hill NJ 08034, (609) 428-1020.

August 12-14, Computer Graphics 80, Birmingham ENGLAND. Computer Graphics 80 will bring together experienced users and specialists to present applications experiences and research findings. In addition to the conference, there will be an equipment exhibition and an animation film festival. To register, contact Paula Stockham, Online, Cleveland Rd, Uxbridge UB8 2DD ENGLAND, phone Uxbridge (0895) 39262.

August 14-24, Electronics/China 80, Guangzhou (Canton) CHINA. This is the first exhibition of U S electronic companies in the People's Republic of China. The United States-China Trade Consultants is the sponsor of the show. Products demonstrated will include circuit components, system elements, test instrumentation, product equipment, and materials. Details are available from Expoconsul Inc, Clapp and Poliak Inc, Princeton-Windsor Office Park, POB 277, Princeton Junction NJ 08550.

August 23-24, Personal Computer Arts Festival, Philadelphia Civic Center. Tutorials, seminars, musical performances, and graphic extravaganzas are the features of this show. Computer musicians and artists have until July 1 to submit material for presentation. Contact PCAF '80 c/o Philadelphia Area Computer Society, POB 1954, Philadelphia PA 19105.

COMING UP IN onComputing

Software in the 80's:
Industry leaders discuss the state of the art and the future of personal computer software.

Texas Instruments Personal Computer:
A look at the 99/4 and its sophisticated speech synthesizer option.

The Lawrence Hall of Science:
Children learn about the exciting world of personal computers in the hills of Berkeley, California.

More About Word Processing

Running an Oil Rig With Personal Computers

InterAction Results

The winning articles for the Winter 1979 onComputing were "Understanding Personal Computer Software" by Louis Frenzel and "A Voice for Bill" by Mark Dahmke. Louis and Mark won first and second prizes of $100 and $50, respectively.

In the Spring 1980 issue, the first and second place winners were "A Beginner's Guide to Programming Languages" by Steve A Hughes and "The Ohio Scientific C4PMF — a Computer for the Real World," by Gregg Williams.

Reader Service

Inquiry		Page
30	Applied Digital Data Systems (ADDS)	5
20	The Bottom Shelf	2
*	BYTE Books	68
240	Creative Discount Software	87
150	Creative Software	55
110	Dennison Kybe Corp	45
80	Dynacomp	31
*	Factory Direct Sales	51
100	Hardside	43
250	Hayden Book Co Inc	91
180	Heath Company	63
120	Hobby World	47
*	Home Computers	93
50	Home Video Magazine	17
70	Infosoft Systems Inc	27
160	Jade Computer	57
215	McClintock Corp	82
210	Microtek C II	
130	MUSE	49
40	NCE	7
90	NRI Schools	37
220	onComputing	81
*	onComputing WATS	66
230	Pan American Electronics (A Radio Shack Auth Sales Ctr)	85
60	Personal & Small Business Computer Show	23
10	Personal Software	1
170	Pickles & Trout	59
260	RITAM International C IV	
190	Serendipity Systems Inc	65
200	Michael Shrayer Software	67
140	Small Business Applications	53
*	Robert Tinney Graphics C III	

interAction

Article Number		Page Number
1	Understanding Floppy Disks	8
2	Computer Animation Helps Children to Learn	18
3	APF Imagination Machine	20
4	Atari 800 Personal Computer	24
5	Nestar Lets Personal Computers Talk to Each Other	28
6	Living With Computers	32
7	Word Processors: A Look at Four Popular Programs	38
8	The Source	60
9	A Do-It-Yourself Text Editor	69
10	A Writer Looks at Word Processors	83

onComputing™

READER SERVICE

Summer 1980

For fastest service transfer mailer label from wrapper to coupon provided below. Requests cannot be honored unless zip code is given. This card valid for 90 days only.

NOTE—If label is missing or defaced fill out coupon carefully—PLEASE PRINT—this is the only way to get requested material to you.

Name _____

(Title) _____

(Company) _____

Address _____

City _____ State _____ Zip Code _____

1	21	41	61	81	101	121	141	161	181	201	221	241	261	281
2	22	42	62	82	102	122	142	162	182	202	222	242	262	282
3	23	43	63	83	103	123	143	163	183	203	223	243	263	283
4	24	44	64	84	104	124	144	164	184	204	224	244	264	284
5	25	45	65	85	105	125	145	165	185	205	225	245	265	285
6	26	46	66	86	106	126	146	166	186	206	226	246	266	286
7	27	47	67	87	107	127	147	167	187	207	227	247	267	287
8	28	48	68	88	108	128	148	168	188	208	228	248	268	288
9	29	49	69	89	109	129	149	169	189	209	229	249	269	289
10	30	50	70	90	110	130	150	170	190	210	230	250	270	290
11	31	51	71	91	111	131	151	171	191	211	231	251	271	291
12	32	52	72	92	112	132	152	172	192	212	232	252	272	292
13	33	53	73	93	113	133	153	173	193	213	233	253	273	293
14	34	54	74	94	114	134	154	174	194	214	234	254	274	294
15	35	55	75	95	115	135	155	175	195	215	235	255	275	295
16	36	56	76	96	116	136	156	176	196	216	236	256	276	286
17	37	57	77	97	117	137	157	177	197	217	237	257	277	297
18	38	58	78	98	118	138	158	178	198	218	238	258	278	298
19	39	59	79	99	119	139	159	179	199	219	239	259	279	299
20	40	60	80	100	120	140	160	180	200	220	240	260	280	300

interAction

Summer 1980

interAction is your direct line to the editor's desk. Each month, the two top rated authors receive bonuses based on your votes. To use this card, refer to the list of authors, titles, and corresponding article numbers on the opposite page. Then rate each article on a scale from 0 to 10 below by circling the appropriate rating number to the right of each article number. Your feedback helps us produce the best possible magazine each month.

..... Rating .....

ARTICLE NUMBER	POOR			FAIR		GOOD		VERY GOOD			EXCELLENT	
1	0	1	2	3	4	5	6	7	8	9	10	
2	0	1	2	3	4	5	6	7	8	9	10	
3	0	1	2	3	4	5	6	7	8	9	10	
4	0	1	2	3	4	5	6	7	8	9	10	
5	0	1	2	3	4	5	6	7	8	9	10	
6	0	1	2	3	4	5	6	7	8	9	10	
7	0	1	2	3	4	5	6	7	8	9	10	
8	0	1	2	3	4	5	6	7	8	9	10	
9	0	1	2	3	4	5	6	7	8	9	10	
10	0	1	2	3	4	5	6	7	8	9	10	
11	0	1	2	3	4	5	6	7	8	9	10	
12	0	1	2	3	4	5	6	7	8	9	10	
13	0	1	2	3	4	5	6	7	8	9	10	
14	0	1	2	3	4	5	6	7	8	9	10	
15	0	1	2	3	4	5	6	7	8	9	10	
16	0	1	2	3	4	5	6	7	8	9	10	
17	0	1	2	3	4	5	6	7	8	9	10	
18	0	1	2	3	4	5	6	7	8	9	10	
19	0	1	2	3	4	5	6	7	8	9	10	
20	0	1	2	3	4	5	6	7	8	9	10	

Comments _____

Place
Stamp
Here

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

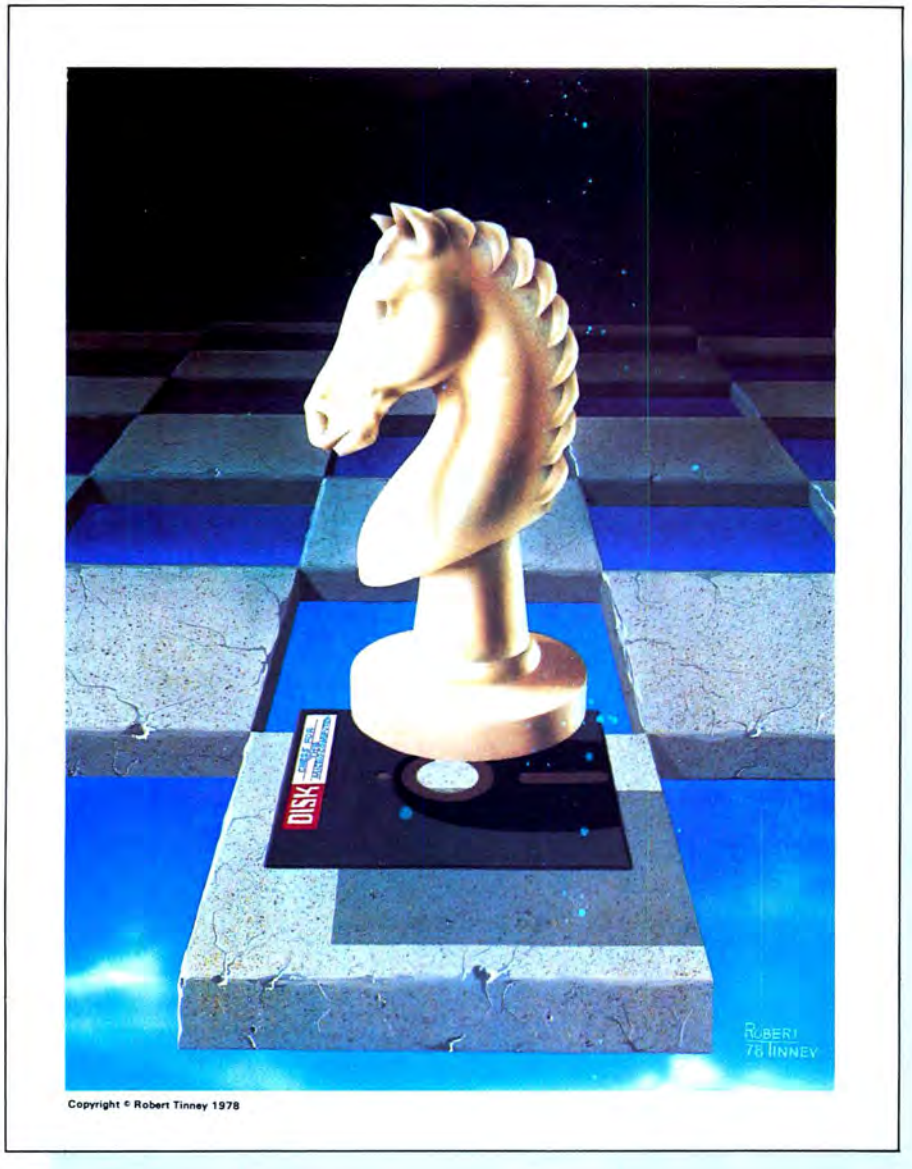
FIRST CLASS Permit Number 7976 New York NY 10001

POSTAGE WILL BE PAID BY ADDRESSEE

onComputing
70 Main St
Peterborough NH 03458
USA

onComputing
POB 1911 GPO
New York NY 10001
USA

onComputing™



Beautiful "Computer Chess" Reproduction—only $4.95!

This dramatic reproduction of the October '78 Byte cover art has been produced with the same care and quality as limited edition prints—yet it is available for the price of a poster.

The overall size is 18" x 22", which includes a 1½" border. It is printed on heavy, 80 lb., matte finish, coated stock, excellent for the finest framing if desired.

The price of this quality reproduction is $4.95, plus $1.00 for mailing tube, handling, and postage. In addition, the artist, Robert Tinney, will select the 100 finest prints from this first edition for his personal signature and number. These 100 signed and numbered prints will be sold on a strictly first-come basis for $14.95 plus $1.00 postage and handling.

See coupon below for ordering.

Send me ____ of the $4.95 posters, and ____ of the $14.95 signed prints. I have included $1.00 for postage and handling.

Please charge this to my Visa/Master Charge: Visa card number: Master Charge no: Card expires:

signature

Name:

Address:

City: State: Zip:

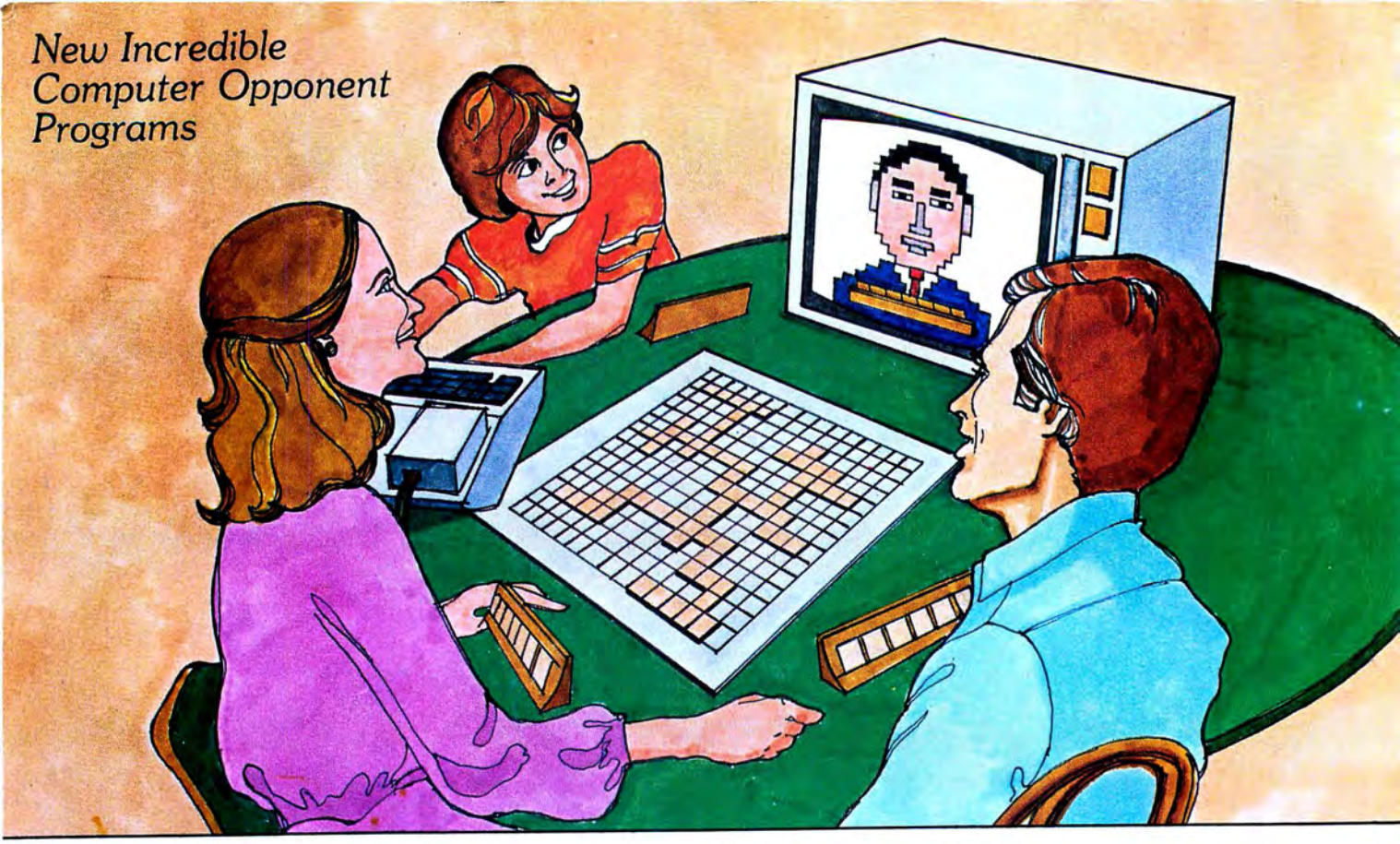
Make check payable to:

robert tinney graphics

P.O. Box 45047

Baton Rouge, LA 70895

New Incredible Computer Opponent Programs



MONTY™

Challenges you to Monopoly* & Scrabble*

MONTY™ brings new life to computer opponent programs. He has been thoroughly tutored in the official rules of America's favorite board games, and plays with consummate skill as he entertains and amuses you with music and delightful visual effects.

"MONTY™ plays Monopoly" and **"MONTY™** plays Scrabble" are available for Apple II and TRS-80 Level II microcomputers. **"MONTY™** plays Scrabble" is also available for CP/M based systems.

These programs, for use with your own game boards and equipment, add more fun and enjoyment to the already exciting games of Monopoly and Scrabble.

MONTY™ is not just another computer game. He represents a new concept in home entertainment which allows the whole family to discover the joy of playing with a computer.

"MONTY™ plays Scrabble"
(Disk interactive - 16K required)

☐ Apple II disk	$29.95
☐ TRS-80 Level II disk	29.95
☐ CP/M 5" disk	29.95
☐ CP/M 8" disk	29.95

"MONTY™ plays Monopoly"
(Note memory requirements)

☐ Apple II cassette (16K-ROM Applesoft)	24.95
☐ Apple II cassette (32K-Cassette Applesoft)	24.95
☐ Apple II disk (32K-ROM Applesoft)	27.95
☐ TRS-80 Level II cassette (16K)	24.95
☐ TRS-80 Level II disk (32K)	24.95

Charge my ☐ VISA ☐ MASTER CHARGE

Credit Card # _____

Exp. Date _____ M.C. Bank # _____

Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Ritam Corporation
P.O. Box 921
Fairfield, Iowa
52556

*Scrabble is a trademark of Selchow & Righter Co.
Monopoly is a trademark of Parker Bros., Inc.
Apple is a trademark of Apple Computer, Inc.
TRS-80 is a trademark of Tandy Corp.
CP/M is a trademark of Digital Research, Inc.
Monty™ is not sponsored or endorsed by Selchow & Righter Co. or Parker Bros.

ORDER NOW TOLL FREE
-VISA or MASTER CHARGE-
(800) 624-7888
(800) 852-7777 (CA)
(800) 824-7919 (HI & AL)
Ask for Operator 975

Circle 260 on inquiry card.